

# Implementierung von Datenbanksystemen 2 (IDBS2)

Sommersemester 2007

Prof. Dr. Erhard Rahm

Universität Leipzig

Institut für Informatik

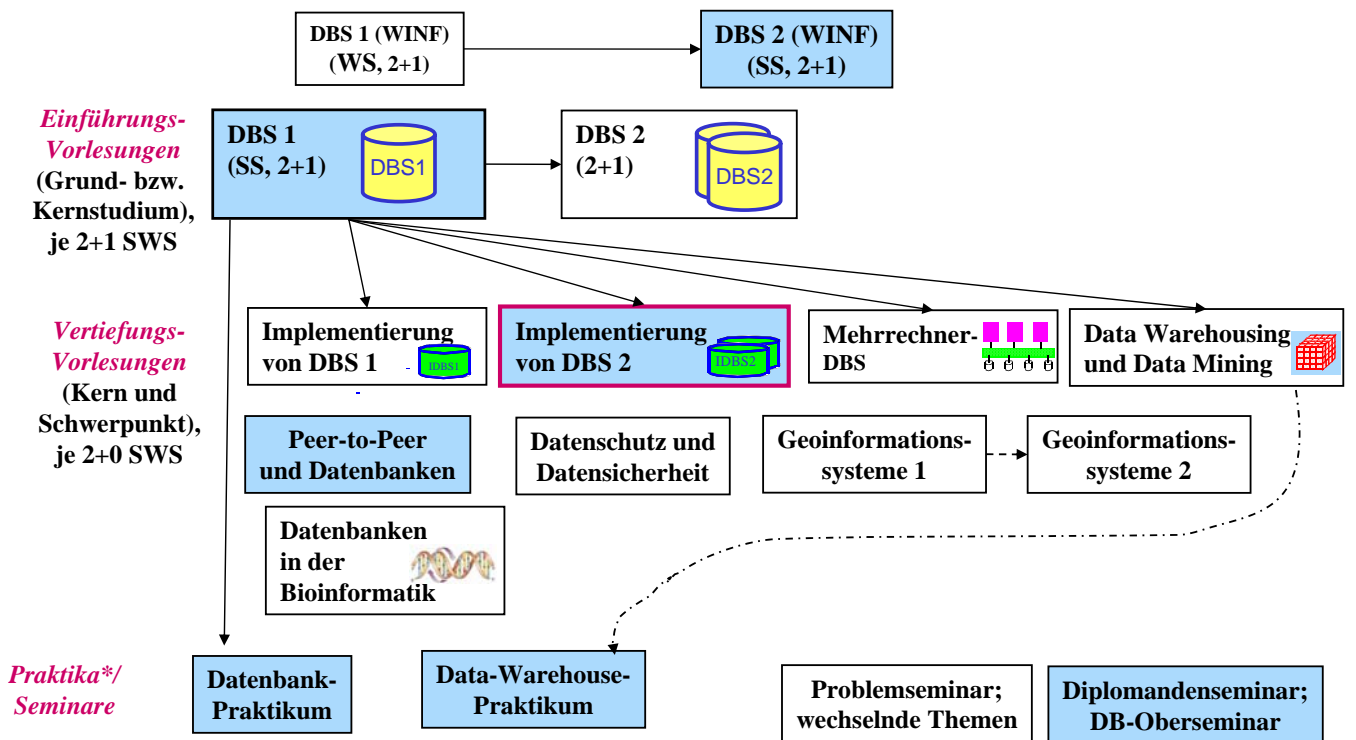
<http://dbs.uni-leipzig.de>



© Prof. E. Rahm

0-1

## Lehrveranstaltungen zu „Datenbanken“ (SS 07)



\* Detaillierter Praktikumsschein wird ausgestellt

© Prof. E. Rahm

0-2



# Leistungsbewertung

- Informatik, Diplom Hauptstudium (ab 5. Semester)
  - Kernfach „Praktische Informatik“
  - Modulklausur im Juli als Teil der Diplomprüfung des Kerngebiets „Praktische Informatik“
  - Kernstudium erfordert Prüfungen zu 6 - 14 SWS Praktische Informatik (von insgesamt 32 SWS)
  - Alternativ: Vorlesung kann als Teil der mündlichen Schwerpunktprüfung gewählt werden
- Informatik Bachelor / Master (alt)
  - APL oder PL (3 Credits / Leistungspunkte)
  - Klausur im Juli
- Neuer Masterstudiengang (ab 2006/07):
  - IDBS2 Teil des Moduls „Anwendungsbezogene Datenbankkonzepte“
  - Klausur im Juli
- Klausur überprüft konzeptionelles Wissen + Anwendungsfälle



## Modulzuordnung für neues Master-Studium (PO2006)

- Abt. Datenbanken bietet folgende Module im Masterstudium an
  - **10-202-2215: Moderne Datenbanktechnologien** - Kleines Modul (Kern, 5 LP)
  - **10-202-2216: Moderne Datenbanktechnologien** - Großes Modul (Vertiefung, 10 LP)
  - **10-202-2213: Anwendungsbezogene Datenbankkonzepte** - Kleines Modul (Kern, 5 LP)
  - **10-202-2214: Anwendungsbezogene Datenbankkonzepte** - Großes Modul (Vertiefung, 10 LP)
- Wahlmöglichkeiten im SS07 für „Anwendungsbezogene Datenbanktechnologien“ aus
  - Implementierung von DBS 2
  - Peer-to-Peer und Datenbanken
  - ggf. Datenbanksysteme 2 / DW-Praktikum
- Musterstudienplan mit Schwerpunkt in Datenbanken/Informationssysteme
  - 1. Semester: Moderne Datenbanktechnologien – Kleines (oder großes) Modul
  - 2. Semester: Anwendungsbezogene Datenbankkonzepte – Kleines (oder großes) Modul
  - 3. Semester: Moderne Datenbanktechnologien – Großes (oder kleines) Modul
  - 4. Semester: **Masterarbeit** / Anwendungsbezogene Datenbankkonzepte – Großes (oder kleines) Modul



# Lernziele der Vorlesung IDBS

- fundierte Kenntnisse der Funktionsweise von Datenbanksystemen
- Implementierungstechniken u.a. zur Sicherstellung einer hohen Performanz der Datenverarbeitung sowie zur Datensicherheit
- IDBS1: Verfahren zur Externspeicher-Nutzung, Verwaltung von Pufferspeichern, Indexstrukturen, Anfrageoptimierung ...
- *IDBS2: Verfahren zur Transaktionsverwaltung: Synchronisation (Concurrency Control), Logging/Archivierung, Recovery ...*
- tiefgehende Kenntnisse wichtig für Datenbank-Administratoren sowie generell für anspruchsvolle DB-Nutzung
- sachkundige Beurteilung von kommerziell verfügbaren DBS
- Verfahren nicht nur für Datenbanksysteme relevant (-> Betriebssysteme, Web-/ Applikations-Server, ...)



## Vorläufige Vorlesungsübersicht

- 1. Einführung: Transaktionsverwaltung, Integritätskontrolle**
- 2. Synchronisation: Grundlagen, Sperrverfahren**
  - Mehrbenutzer-Anomalien
  - Serialisierbarkeit
  - Sperrverfahren: 2PL, Hierarchische Sperrverfahren
  - Konsistenzstufen
  - Deadlock-Behandlung
- 3. Synchronisation: Weitere Verfahren, Leistungsbewertung**
  - Optimistische Verfahren
  - Zeitstempel- und Mehrversionen-Verfahren
  - Spezialverfahren für B\*-Bäume und „High Traffic“-Elemente
  - Leistungsanalyse und Lastkontrolle



# Vorlesungsübersicht (2)

## 4. Logging und Recovery: Grundlagen

- Begriffe und Annahmen, Fehlermodell
- Logging-Verfahren
- Klassifikation von Recovery-Strategien

## 5. Crash- und Medien-Recovery

- Crash-Recovery
- Platten-Recovery

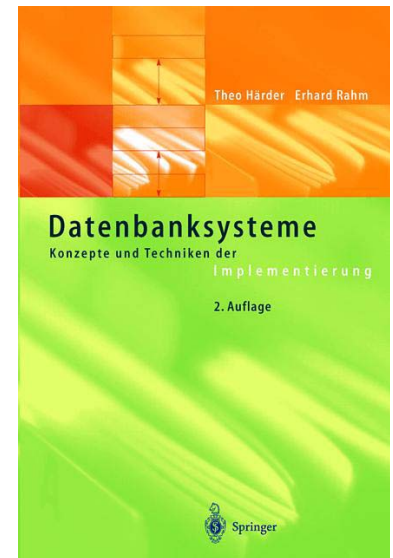
## 6. Transaktionskonzept: Weiterentwicklungen

- Geschachtelte Transaktionen
- Transaktionsketten (Sagas)




## Literatur

- Härder, T., Rahm, E.: *Datenbanksysteme - Konzepte und Techniken der Implementierung*. Springer-Verlag, 2. Auflage 2001 (Kap. 1 und 13 online)
- Weitere Lehrbücher
  - Garcia-Molina, H., Ullman, J.D., Widom, J.: *Database System Implementation*. Prentice Hall, 2000
  - Saake, G., Heuer, A.: *Datenbanken: Implementierungstechniken*, MITP-Verlag, 1999
- Forschungsergebnisse
  - Tagungsbände: VLDB (jährliche Konferenz "Very Large Data Bases"), SIGMOD (Konferenz der ACM Special Interest Group on Management of Data), IEEE Data Engineering, EDBT, BTW ...
  - Zeitschriften: VLDB Journal (Very Large Data Bases), ACM TODS (Transactions on Database Systems), Datenbank-Spektrum ...
- DBLP-Portal: <http://dblp.uni-trier.de>  
(>860.000 Referenzen, viele Links auf Volltexte, Homepages etc.)



# Online-Übungen

- **LOTS** (Leipzig Online Test System), <http://lots.uni-leipzig.de>  
– Kennung idbs2#07



## Leipzig Online-Test-System

UNIVERSITÄT LEIPZIG  
Fakultät für Mathematik und Informatik  
Institut für Informatik  
Abteilung Datenbanken

Home | Registrierung | Impressum

### Login

Username:

Password:

[Passwort vergessen?](#)

### Gast Login


Sie können sich als Gast einloggen, um LOTS ohne vorherige Anmeldung zu testen.  
Der Gast Account ist auf 30 min Benutzung und im Funktionsumfang beschränkt.  
Bei weiterem Interesse sollten Sie sich als Gast [registrieren](#).

Viel Spass!

Ihr LOTS Team

### System Info

# Benutzer: 31

 05.02.2007  
12:15:21

### News

Bitte loggen Sie sich ein, um personalisierte News lesen zu können.



## LOTS-Online-Übungen

### Aufgabe 3 (Nested-Block-Join)

Für den Gleichverbund zwischen R und S (je 100.000 Sätze, Blockungsfaktor 100) soll ein Nested-Block-Join genutzt werden. Welche Kombinationen zwischen verfügbarer Hauptspeichergröße M und Anzahl erreichbarer Plattenzugriffe (ohne Schreiben des Resultats) treffen zu?

0 von 2

- M=1001; 1 Million Plattenzugriffe
- M=5001; 2000 Plattenzugriffe
- M=1001; 2000 Plattenzugriffe
- M=501; 3000 Plattenzugriffe



### Aufgabe 4 (Hash-Join)

Markieren Sie die zutreffenden Aussagen.

0 von 2

- Durch die Nutzung von Bitvektoren lässt sich der Umfang der Hash-Tabellen kleiner halten
- Hash-Joins sollten nur eingesetzt werden, wenn die kleinere Eingabetabelle im Hauptspeicher gespeichert werden kann
- Hash-Joins können effektiv zur Realisierung von Mehr-Wege-Joins genutzt werden
- Hash-Join-Verfahren eignen sich zur Beantwortung von Equi-Join-Anfragen



# Neu: Erklärungsfunktion

## Aufgabe 3 (Nested-Block-Join)

0 von 2

Für den Gleichverbund zwischen R und S (je 100.000 Sätze, Blockungsfaktor 100) soll ein Nested-Block-Join genutzt werden. Welche Kombinationen zwischen verfügbarer Hauptspeichergröße M und Anzahl erreichbarer Plattenzugriffe (ohne Schreiben des Resultats) treffen zu?

M=1001; 1 Million Plattenzugriffe



M=5001; 2000 Plattenzugriffe

Erklärung

M=1001; 2000 Plattenzugriffe

Erklärung

komplettes Einlesen von R: 1000 Plattenzugriffe (Blöcke). Danach sukzessives Lesen jedes S-Blocks (1000-mal) -> 1000 Plattenzugriffe. Zusammen sind das 2000 Plattenzugriffe.

M=501; 3000 Plattenzugriffe

Erklärung

Einlesen der ersten 500 R-Seiten; sukzessives Einlesen je 1 S-Seite (1000-mal) -> 1500 Plattenzugriffe (Blöcke). Analog für die zweiten 500 R-Seiten. -> 2=1500=3000 Plattenzugriffe



## LOTS-Nutzung vs. Klausurerfolg

