Seminar:
Advanced Topics in Databases
Präsentation der Themen

Felix Naumann

Alexander Albrecht

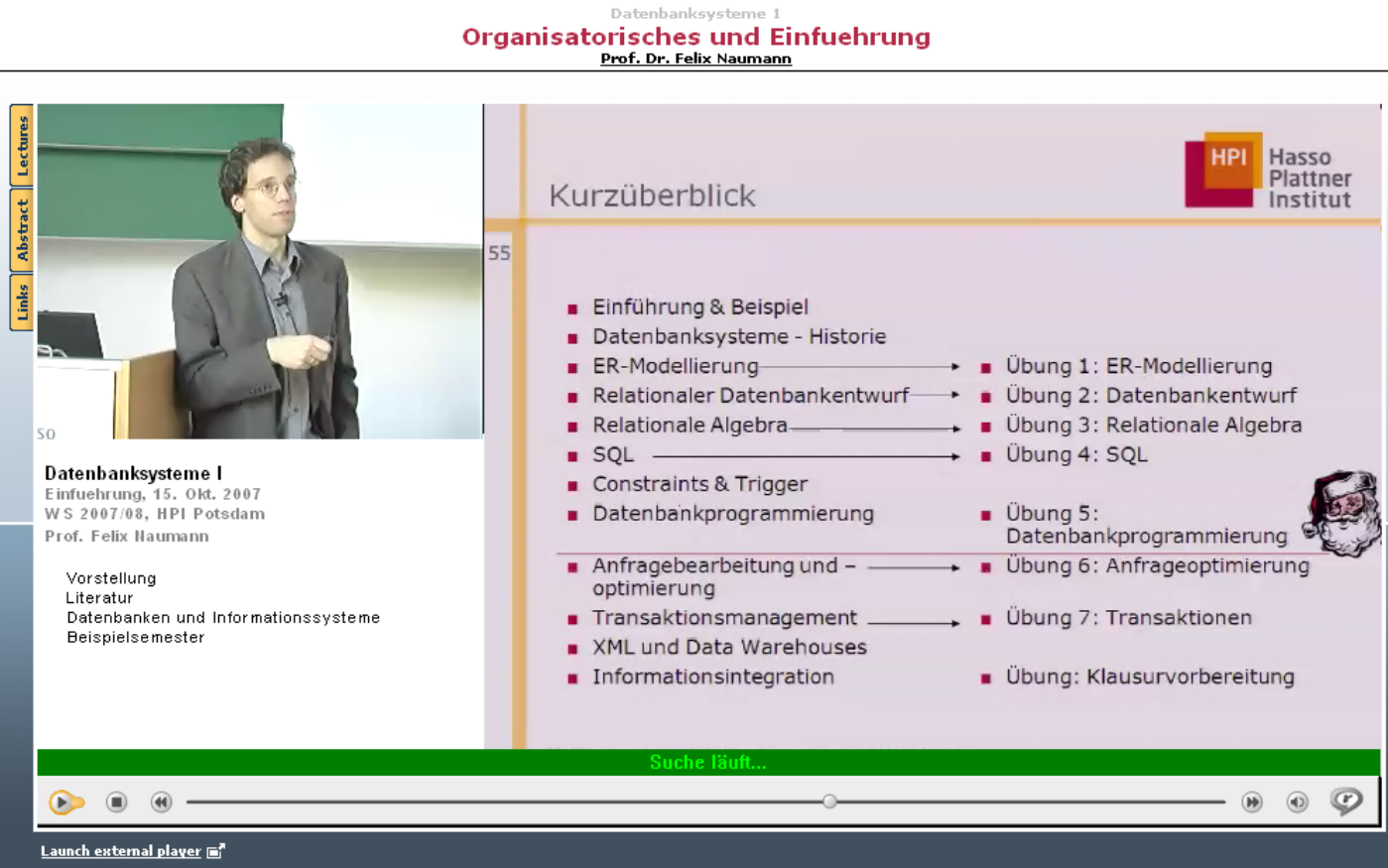
Jens Bleiholder

Hasso-Plattner-Institut für Softwaresystemtechnik GmbH
FB Informationssysteme
Prof.-Dr.-Helmert-Str. 2-3, D-14482 Potsdam

Überblick

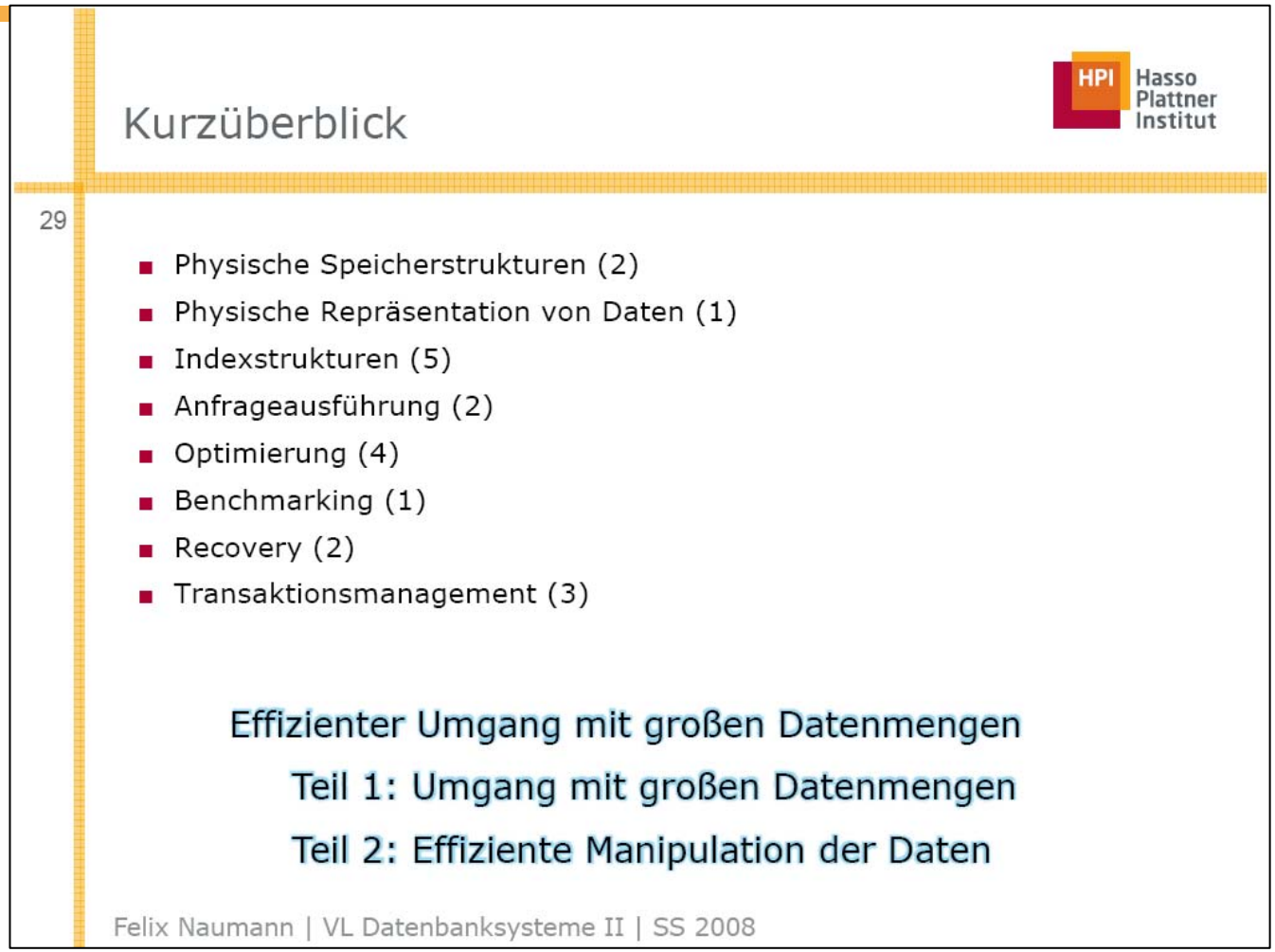
- **Einführung und Motivation**
- **Vorstellung der Themen**
  - Query Processing
  - Data Storage and Access Methods
  - Transaction Management
  - Extensible Systems
  - Database Evolution
  - Data Warehousing
  - Data Mining
  - Web Services and Databases
  - Stream-Based Data Management
- **Organisatorisches**

Voraussetzungen – VL DBS I



- http://www.hpi.uni-potsdam.de/naumann/lehre/ws_0607/dbsi.html
- http://www.tele-task.de/page42_mode1_series617.html

Voraussetzungen – VL DBS II



- http://www.hpi.uni-potsdam.de/naumann/lehre/ss_08/dbsii0.html
- http://www.tele-task.de/page42_mode1_series645.html

Seminar - Advanced Topics in Databases

- Ziel des Seminars: Tieferer Einblick in die wichtigen Themen der Datenbank-Forschung
- Nicht nur hochaktuelle sondern auch grundlegende Themen werden behandelt
- "Readings in Database Systems" von Joseph M. Hellerstein und Michael Stonebraker bilden die Grundlage
  - Exzellente Sammlung bedeutender Paper im Bereich Datenbanken ("best of")
  - http://redbook.cs.berkeley.edu/
    http://www.amazon.de/dp/0262693143/

Überblick

- **Einführung und Motivation**
- **Vorstellung der Themen**
  - Query Processing
  - Data Storage and Access Methods
  - Transaction Management
  - Extensible Systems
  - Database Evolution
  - Data Warehousing
  - Data Mining
  - Web Services and Databases
  - Stream-Based Data Management
- **Organisatorisches**

Query Processing

- **Access Path Selection in a Relational Database Management System**
- SIGMOD **1979**
- P. Selinger, M. Astrahan, D. Chamberlin, R. Lorie, T. Price

- Beschreibt, wie in System R, einem der ersten relationalen DBMSe von IBM, ausgewählt wird, wie auf die Daten auf der Platte zugegriffen wird. Gegeben ist eine SQL Anfrage, geliefert wird ein Zugriffsplan auf die Daten, der möglichst wenig Kosten verursacht.

Query Processing

- **Join Processing in Datanbase Systems with Large Main Memories**
- TODS 11(3), **1986**
- L. Shapiro

- Nimmt an, dass es zukünftig Systeme mit genügend großem Hauptspeicher geben wird, um Hash-Techniken zur Berechnung eines Joins verwenden zu können. Führt den Hash-Join ein.

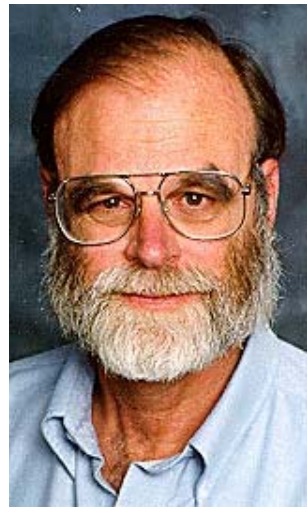
Data Storage and Access Methods

- **The R*-tree: An Efficient and Robust Access Method for Points and Rectangles**
- SIGMOD **1990**
- N. Beckmann, H.-P. Kriegel, R. Schneider, B. Seeger

- R-trees sind Speicherstrukturen um auf Raumdaten (Punkte, Rechtecke, etc.) zuzugreifen. Basierend auf dem B+-tree werden in diesem Artikel mehrere Varianten von R-trees miteinander verglichen.

Data Storage and Access Methods

- **The Five-Minute Rule Ten Years Later, and Other Computer Storage Rules of Thumb**
- SIGMOD Record 26(4), **1997**
- J. Gray, G. Graefe



- Wann lohnt es sich, eine Seite im Hauptspeicher zu halten, anstatt sie jedes Mal von Platte zu lesen? Wenn sie alle 5 Minuten benötigt wird. Welche Regeln gibt es noch, um zu entscheiden wie ich meine Speicherhierarchie aufbaue?

Transaction Management

- **On Optimistic Methods for Concurency Control**
- VLDB **1979**
- H. Kung, J. Robinson

- Der Artikel beschreibt eine Methode ohne Locking verteilt und parallel von mehreren Prozessen aus, auf Daten zuzugreifen.

Transaction Management

- **Efficient Locking for Concurrent Operations on B-trees**
- TODS 6(4), **1981**
- P. Lehman, S. Yao

- Wie kann man sicherstellen, dass bei gleichzeitigem Zugriff mehrerer Transaktionen auf denselben B-Baum keine Inkonsistenzen auftreten? Für diesen Zweck wird ein Locking-Protokoll vorgestellt.

Extensible Systems

- **Generalized Search Trees for Database Systems**
- VLDB **1995**
- J. Hellerstein, J. Naughton, A. Pfeffer

- Für unterschiedliche Domänen und Anfragetypen gibt es spezialisierte Suchstrukturen (B-Bäume, R-Bäume, etc.). Mit dem GiST wird eine allgemeine Suchstruktur vorgestellt, die auf verschiedenste Art angepasst und spezialisiert werden kann.

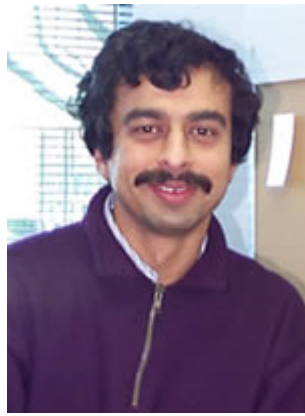
Extensible Systems

- **Grammar-like Functional Rules for Representing Query Optimization Alternatives**
- SIGMOD **1988**
- G. Lohman

- Ein Anfrageoptimierer erstellt aus der SQL Anfrage den auszuführenden optimalen Ausführungsplan, üblicherweise mit Hilfe von Transformationsregeln. Der Artikel beschreibt eine Methode schnell mehrere alternative Pläne zu erzeugen.

Database Evolution

- **AutoAdmin 'What-if' Index Analysis Utility**
- SIGMOD, 1998
- Surajit Chaudhuri, Vivek R. Narasayya



- **Automatisierte Index-Auswahl in Datenbanken:** Das AutoAdmin Projekt präsentierte das erste Tool, um ausgehend von einer Menge SQL-Anweisungen, dem Workload, die Kosten für unterschiedliche Index-konfigurationen abschätzen zu können. Das Tool nutzt dabei den Datenbank-Optimierer für die Kosten-abschätzung. Prototyped wurde für Microsoft SQL Server 7.0.

Database Evolution

- **Applying Model Management to Classical Meta Data Problems**
- CIDR, 2003
- Philip A. Bernstein

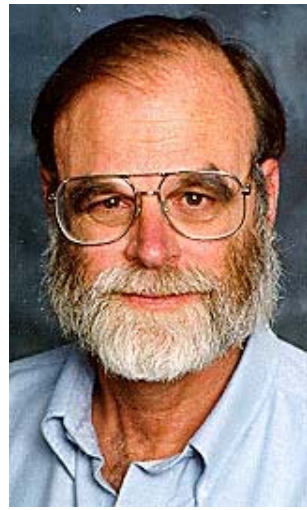
- **Metadata Management:** Beim Austausch von Daten, Weiterverarbeiten von (XML-)Nachrichten, Objekt-relationalen Mappings etc. treten klassische Metadaten Management Probleme auf, z.B. Schema Integration. Das Paper beschreibt Model Management und dessen Anwendung als neuen Ansatz um diese Probleme anzugehen.
- **On top**: Rondo - A Programming Platform for Generic Model Management. SIGMOD, 2003.

- **Improved Query Performance with Variant Indexes**
- SIGMOD, 1997
- Patrick E. O'Neil, Dallan Quass

- **Indexing:** Das read-only Umfeld in einem Data-warehouse ermöglicht komplexere Index-Verfahren, z.B. bzgl. Materialisierung. Das Paper stellt neue Index-Strukturen vor. Diese neuen Verfahren ermöglichen es, im Vergleich zu klassischen Index-Strukturen (B-Bäume, Bitmap), OLAP-Anfragen schneller auszuwerten. Dabei wird die Groupset Index-Struktur genauer beschrieben, die insbesondere für OLAP-Anfragen mit Aggregation und Gruppierung gut geeignet ist.

Data Warehousing

- **Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals**
- *Data Mining and Knowledge Discover*, Vol. 1, No. 1, 1997
- Jim Gray et al.



- **Online Analytical Processing (OLAP):** Bei der Datenanalyse werden Daten oft gleichzeitig nach mehreren Dimensionen gruppiert, z.B. nach Zeit (Tag/Woche/Jahr) oder Ort (Stadt/Land/Kontinent), und aggregiert (roll-up), z.B. Umsätze oder Lagerbestände. Klassisches SQL kann das nicht leisten. Das Paper erweitert SQL um den **Data Cube** und **Roll-Up** Operator für die schnelle und mehrdimensionale Datenanalyse.

Data Mining

- **BIRCH: An Efficient Data Clustering Method for Very Large Databases**
- SIGMOD, 1996
- Tian Zhang, Raghu Ramakrishnan, Miron Livny

- **Data Clustering** entdeckt in einer Menge von Daten dicht beieinander liegenden Datenpunkte (Cluster). Dabei verwendet BIRCH hierarchisches Clustering, um in sehr großen Datenmengen Cluster für metrische, mehr-dimensionale Datenpunkte zu finden. In dem Paper wird der CF Baum als neue Datenstruktur vorgestellt, um hierarchisches Clustering großer Datenmengen effizient im vergleichsweise kleinen Hauptspeicher zu ermöglichen.



Data Mining

- **Fast Algorithms for Mining Association Rules in Large Databases**
- VLDB, 1994
- Rakesh Agrawal, Ramakrishnan Srikant

- **Association Rules:** In diesem Paper werden zwei effiziente Algorithmen vorgestellt, die (unbekannte) Zusammenhänge in transaktionalen Daten entdecken. Zusammenhänge werden dabei als Assoziationsregeln formuliert, X=>Y, wobei X und Y Mengen von Items sind und es gilt: Transaktionen, die X beinhalten, enthalten in c% aller Transaktionen auch Y.
- Für Studenten, die **nicht** an den vorangegangenen '*Beauty is our Business*' Seminaren teilgenommen haben.

- **The Anatomy of a Large-Scale Hypertextual Web Search Engine**
- Computer Networks, 30(1-7), 1998
- Sergey Brin, Lawrence Page

- “In this paper, we present Google, a prototype of a large-scale search engine which makes heavy use of the structure present in hypertext. Google is designed to crawl and index the Web efficiently and produce much more satisfying search results than existing systems …”

Stream-Based Data Management

- **The Design and Implementation of a Sequence Database System**
- VLDB, 1996
- Praveen Seshadri, Miron Livny, Raghu Ramakrishnan

- **Sequence data:** Daten in einer Relation können oft als Sequenz betrachtet werden, denen eine logische Ordnung zugrunde liegt, beispielsweise können Daten nach der Zeit geordnet sein (Verkehrsbeobachtungs-system). In relationalen Datenbanken werden Daten ausschließlich als Menge betrachtet, nicht als Sequenz. Das Paper beschreibt die Erweiterung eines relationalen Datenbanksystems, um Sequenzen zu unterstützen und die Interaktion von Sequence Queries (SEQULN) und Relational Queries (SQL) zu ermöglichen.

- **Eddies: Continuously Adaptive Query Processing**
- SIGMOD, 2000
- Ron Avnur, Joseph M. Hellerstein

- **Anfrageverarbeitung:** In der klassischen Anfrage-verarbeitung wird der Anfrageplan einmalig erstellt und bleibt während der Ausführung unverändert. Das führt zu Problemen, sobald sich die Laufzeitumgebung während der Anfrageverarbeitung ändert (förderierte Datenbanken etc.). In diesem Paper wird mit Eddy ein Mechanismus zur Anfrageverarbeitung vorgestellt, der während der Laufzeit die Operatoren im Anfrageplan reorganisiert. Das Reorganisieren geschieht dabei unter anderem mit Hilfe einfacher maschineller Lernverfahren.

Überblick

- **Einführung und Motivation**
- **Vorstellung der Themen**
  - Query Processing
  - Data Storage and Access Methods
  - Transaction Management
  - Extensible Systems
  - Database Evolution
  - Data Warehousing
  - Data Mining
  - Web Services and Databases
  - Stream-Based Data Management
- **Organisatorisches**

Organisatorisches

- Feste Termine
  - Dienstag um 9:15 Uhr
  - Raum A-1.1
  - Nächster Termin (28. Oktober) – Einführung LaTeX
- Pro Termin
  - Zwei Vorträge / zwei Themen
  - Diskussion über die Themen

Organisatorisches

- Zu einem ausgewählten Thema
  - Vortrag, ca. 30 Minuten und 15 Minuten Diskussion
  - Ausarbeitung, ca. 8-10 Seiten, in LaTeX

- Zu den nicht gewählten Themen
  - Einarbeitung in die Themen vor jedem Seminartermin (Paper lesen)
  - Mitarbeit wird bei der Bewertung berücksichtigt

- Anwesenheit an allen Terminen!
- Vortrag, Ausarbeitung und Beteiligung im Seminar werden benotet!

Ablauf

- Paper auswählen
  - Geordnete Themenwunschliste bis spätestens **23. Oktober** per E-Mail an Jens Bleiholder mit mind. 3 Themen aus mind. 3 der vorgestellten Themengebiete. Die Reihenfolge ist wichtig!
- Themenvergabe & Terminbekanntgabe am **24. Oktober**
  - Benachrichtigung erfolgt per E-Mail und im WWW
- Paper lesen
  - Ab 24. Oktober

Betreuung

- 1. Treffen mit Betreuer
  - Inhalte besprechen, ca. 2 Wochen vor Vortragstermin
- 2. Treffen mit Betreuer
  - Vortrag/Folien besprechen, ca. 1 Woche vor Vortragstermin
- 3. Treffen mit Betreuer
  - Ausarbeitung besprechen, ca. 2 Wochen vor Abgabe
- Abgabe der Ausarbeitung am **3. März 2009**

Thank you for your interest!