

Prof. Dr.-Ing. Dr. h. c. Theo Härder  
 AG Datenbanken und Informationssysteme  
 Zi. 36/330, Tel.: 0631-205-4030  
 E-Mail: haerder@informatik.uni-kl.de  
 http://www.dvs.informatik.uni-kl.de/

## Datenbankanwendung

Wintersemester 2006/2007

TU Kaiserslautern  
 Fachbereich Informatik  
 Postfach 3049  
 67653 Kaiserslautern

### Vorlesung:

Beginn: 25.10.2006

Ort: 11-262

Zeit: Mo., 10.00-11.30 Uhr  
 und

Ort: 48-210

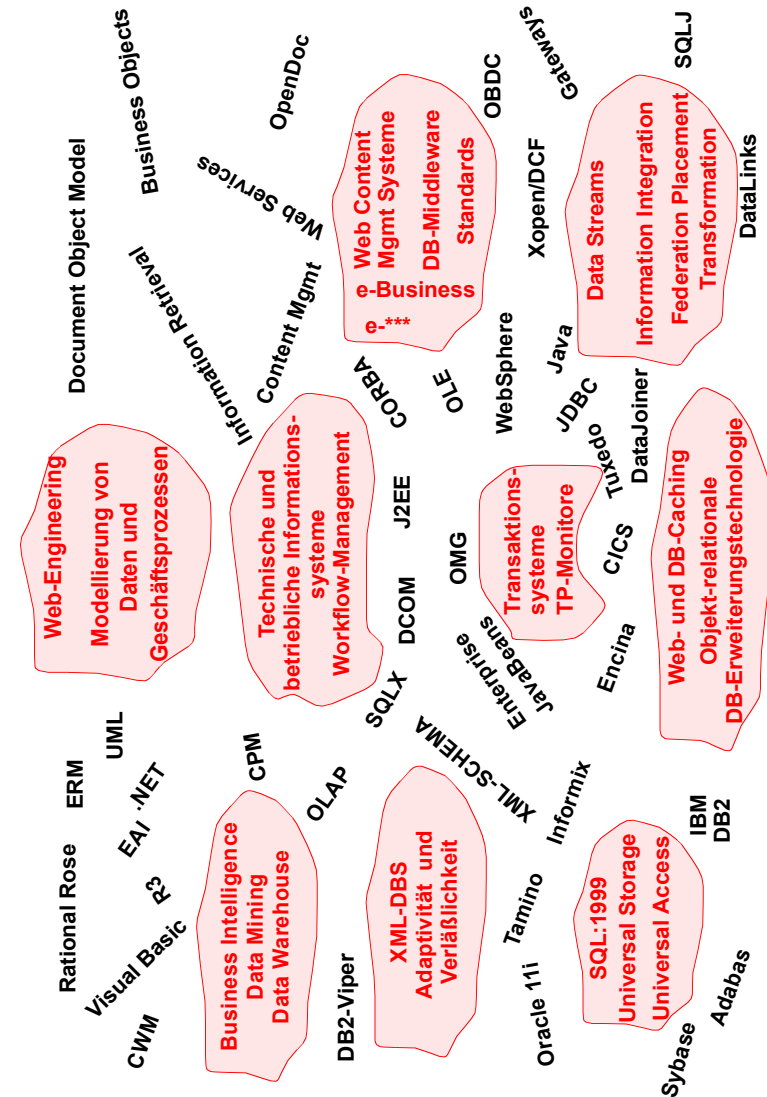
Zeit: Mi., 11.45-13.15 Uhr

### Übung:

Ort: 13-222

Zeit: Do., 15.30-17.00 Uhr

## Weltbild des Lehrgebietes Informationssysteme



## Ziele

- **Vermittlung von Grundlagen- und Methodenwissen<sup>1</sup> zur Anwendung von Datenbanksystemen; Erwerb von Fähigkeiten und Fertigkeiten für DB-Administrator und DB-Anwendungsentwickler**
  - Entwurf, Aufbau und Wartung von Datenbanken sowie Programmierung und Übersetzung von DB-Programmen, insbesondere auf der Basis von
    - Relationenmodell und SQL
    - objektorientierten und objekt-relationalen Datenmodellen mit Bezug auf die Standards ODMG und SQL:1999
  - Sicherung der DB-Daten und der Abläufe von DB-Programmen
    - Transaktionsverwaltung
    - Synchronisation
    - Fehlerbehandlung (Logging und Recovery)
    - Semantische Integrität, aktive DB-Mechanismen
    - Datenschutz und Zugriffskontrolle
- **Voraussetzungen für Übernahme von Tätigkeiten:**
  - Entwicklung von datenbankgestützten Anwendungen
  - Nutzung von Datenbanken unter Verwendung von (interaktiven) Datenbanksprachen
  - Systemverantwortlicher für Datenbanksysteme, insbesondere Unternehmens-, Datenbank-, Anwendungs- und Datensicherungsadministrator

1. Grundlagenwissen ist hochgradig allgemeingültig und nicht von bestimmten Methoden abhängig. Die Halbwertszeit ist sehr hoch. Methodenwissen muß ständig an die aktuelle Entwicklung angepaßt werden. In der Informatik haben sich die entscheidenden Methoden alle 8-10 Jahre erheblich geändert. Werkzeugwissen ist methodenabhängig. Werkzeuge haben in der Informatik oft nur eine Lebensdauer von 2-3 Jahren.

## ÜBERSICHT (vorl.)

### 0. Übersicht und Motivation

- Zusammenfassung: Relationenmodell
- Einige künftige Entwicklungen

### 1. Anforderungen und Beschreibungsmodelle

- Anforderungen an DBS
- Aufbau von DBS
- Beschreibungsmodelle (Fünf-Schichten-Modell, Drei-Ebenen-Beschreibungsarchitektur)

### 2. Logischer DB-Entwurf

- Konzeptioneller DB-Entwurf
- Normalformenlehre (1NF, 2NF, 3NF, 4NF)
- Synthese von Relationen

### 3. Tabellen und Sichten

- Datendefinition von SQL-Objekten
- Schemaevolution
- Indexstrukturen
- Sichtenkonzept

### 4. Anwendungsprogrammierschnittstellen

- Kopplung mit einer Wirtssprache
- Übersetzung und Optimierung von DB-Anweisungen
- Eingebettetes / Dynamisches SQL, PSM
- CLI, JDBC und SQLJ

## ÜBERSICHT (2)

### 5. Transaktionsverwaltung

- Transaktionskonzept, Ablauf von Transaktionen
- Commit-Protokolle
  - für zentralisierten Ablauf
  - für verteilten Ablauf mit zentralisierter Kontrolle: 2PC und Optimierungen

### 6. Serialisierbarkeit

- Anomalien beim Mehrbenutzerbetrieb
- Theorie der Serialisierbarkeit
  - Final-State-Serialisierbarkeit, Sichtenserialisierbarkeit
  - Konfliktserialisierbarkeit
- Klassen von Historien

### 7. Synchronisation – Algorithmen

- Sperrprotokolle (Deadlocks, S2PL und SS2PL)
- Nicht-sperrende Protokolle (Zeitstempel, SGT)
- Optimistische Synchronisation (BOCC, FOCC)

### 8. Sperrverfahren – Implementierung und Analyse

- Zweiphasen-Sperrprotokolle
  - RUX-Protokoll
  - hierarchische Verfahren
- Konsistenzebenen
- Optimierungen (Mehrversions-, Prädikats-, Objektsperren, spezielle Protokolle)
- Leistungsbewertung und Lastkontrolle

## ÜBERSICHT (3)

### 9. Logging und Recovery

- Fehlermodell und Recovery-Arten
- Logging-Strategien
- Recovery-Konzepte – Abhängigkeiten
- Sicherungspunkte
- Transaktions-, Crash- und Medien-Recovery

### 10. Integritätskontrolle und aktives Verhalten

- Semantische Integritätskontrolle
- Regelverarbeitung in DBS, Trigger-Konzept von SQL
- Definition und Ausführung von ECA-Regeln

### 11. Datenschutz und Zugriffskontrolle

- Technische Probleme des Datenschutzes
- Konzepte der Zugriffskontrolle, Zugriffskontrolle in SQL
- Sicherheitsprobleme in statistischen Datenbanken

### 12. Objektorientierung und Datenbanken

- Beschränkungen klassischer Datenmodelle
- Grundkonzepte der Objektorientierung
- SQL:1999 – Neue Funktionalität
  - ORDBS: Anforderungen, Architekturvorschläge
  - Erhöhung der Anfragemächtigkeit, Rekursion

### 13. Große Objekte

- Anforderungen und Verarbeitung mit SQL
- Lokator-Konzept, Speicherungsstrukturen, . . .

## LITERATURLISTE

- Connolly, T., Begg, C.: Database Systems – A Practical Approach to Design, Implementation, and Management, 4th Edition, Addison Wesley, 2005
- Elmasri, R., Navathe, S. B.: Grundlagen von Datenbanksystemen, Ausgabe Grundstudium, 3. Auflage, Pearson Studium, 2005
- Härder, T., Rahm, E.: Datenbanksysteme – Konzepte und Techniken der Implementierung, Springer-Verlag, Berlin, 2001
- Hoffer, J., Prescott, M., McFadden, F.: Modern Database Management (International Edition), 7. Auflage, Pearson Studium, 2005
- Kemper, A., Eickler, A.: Datenbanksysteme – Eine Einführung, 6. Auflage, Oldenbourg-Verlag, 2006
- Kifer, M., Bernstein, A., Lewis, P. M.: Database Systems – An Application-Oriented Approach, 2nd Edition, Pearson International Edition, 2006
- Weikum, G., Vossen, G.: Transactional Information Systems, Morgan Kaufmann Publishers, San Francisco, CA, 2002

### ZEITSCHRIFTEN:

- ACM TODS Transactions on Database Systems, ACM Publikation (vierteljährlich)
- Information Systems Pergamon Press (6-mal jährlich)
- The VLDB Journal VLDB Foundation (vierteljährlich)
- Informatik – Forschung und Entwicklung Springer Verlag (vierteljährlich)
- ACM Computing Surveys ACM-Publikation (vierteljährlich)

### TAGUNGSBÄNDE:

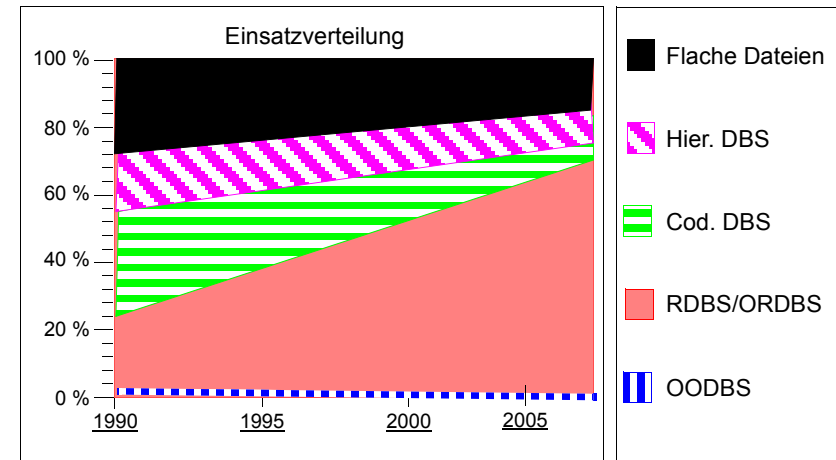
- SIGMOD Tagungsbände, jährliche Konferenz der ACM Special Interest Group on Management of Data
- VLDB Tagungsbände, jährliche Konferenz „Very Large Data Bases“
- ICDE Tagungsbände, jährliche Konferenz „Int. Conf. on Data Engineering“
- BTW Tagungsbände der alle 2 Jahre stattfindenden Tagungen „Datenbanksysteme für Business, Technologie und Web“ der GI, und weitere Tagungen innerhalb des GI-FB „DBIS“

und viele weitere Konferenzreihen

## Verteilung von DBS und Dateien

### • Es gibt verschiedenartige Datenmodelle und die sie realisierenden DBS

- relational und objekt-relational (RDBS/ORDBS)
- hierarchisch (DBS nach dem Hierarchiemodell)
- netzwerkartig (DBS nach dem Codasyl-Standard)
- objektorientiert (OODBS)



### • Künftige DBS

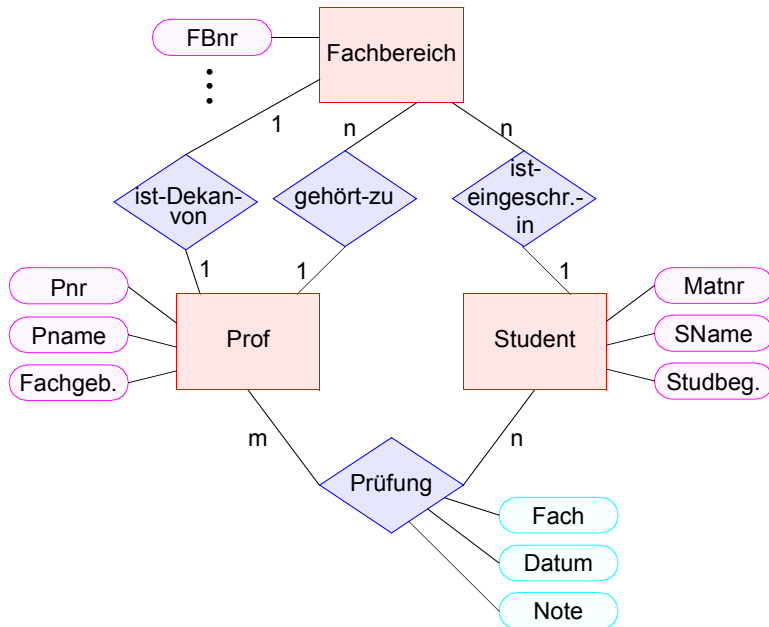
- Aufstellung berücksichtigt nur strukturierte Daten. 85% der weltweit verfügbaren Daten aber sind semi- oder unstrukturiert (Internet, wiss. Aufzeichnungen und Experimente usw.)
- SQL-XML-DBS, XML-SQL-DBS, native XML-DBS

### • Voraussetzung für die Vorlesung: Beherrschung von

- Informationsmodellen (erweitertes ER-Modell)
- Relationenmodell und Relationenalgebra
- SQL-92 als Standardsprache

## Informationsmodellierung

### • Entity/Relationship-Diagramm einer Beispiel-Miniwelt



- Spezifikation benutzerdefinierter Beziehungen (rein struktureller Natur)
- Klassifikation der Beziehungstypen (1:1, 1:n, n:m)

#### → relativ semantikarme Darstellung von Weltausschnitten

- deshalb Verfeinerungen/Erweiterungen durch
  - **Kardinalitätsrestriktionen** ([1,1]:[1,10], [0:1]:[0:\*])
  - **Abstraktionskonzepte**  
(Klassifikation/Instantiierung, Generalisierung/Spezialisierung, Element-/Mengen-Assoziation, Element-/Komponenten-Aggregation)

## Relationenmodell – Beispiel

### DB-Schema

FB		
<u>FBNR</u>	FBNAME	DEKAN

PROF			
<u>PNR</u>	PNAME	FBNR	FACHGEB

STUDENT			
<u>MATNR</u>	SNAME	FBNR	STUDBEG

PRÜFUNG				
<u>PNR</u>	<u>MATNR</u>	FACH	DATUM	NOTE

### Ausprägungen

FB	<u>FBNR</u>	FBNAME	DEKAN
	FB 9	WIRTSCHAFTSWISS	4711
	FB 5	INFORMATIK	2223

PROF	<u>PNR</u>	PNAME	FBNR	FACHGEB
	1234	HÄRDER	FB 5	DATENBANKSYSTEME
	5678	WEDEKIND	FB 9	INFORMATIONSSYSTEME
	4711	MÜLLER	FB 9	OPERATIONS RESEARCH
	6780	NEHMER	FB 5	BETRIEBSSYSTEME

STUDENT	<u>MATNR</u>	SNAME	FBNR	STUDBEG
	123 766	COY	FB 9	1.10.00
	225 332	MÜLLER	FB 5	15.04.97
	654 711	ABEL	FB 5	15.10.99
	226 302	SCHULZE	FB 9	1.10.00
	196 481	MAIER	FB 5	23.10.00
	130 680	SCHMID	FB 9	1.04.02

PRÜFUNG	<u>PNR</u>	<u>MATNR</u>	FACH	PDATUM	NOTE
	5678	123 766	BWL	22.10.03	4
	4711	123 766	OR	16.01.02	3
	1234	654 711	DV	17.04.03	2
	1234	123 766	DV	17.04.03	4
	6780	654 711	SP	19.09.03	2
	1234	196 481	DV	15.10.03	1
	6780	196 481	BS	23.10.03	3

## Relationenmodell – Beispiel (2)

### • Deskriptive DB-Sprachen

- hohes Auswahlvermögen und Mengenorientierung
- leichte Erlernbarkeit auch für den DV-Laien
- RM ist symmetrisches Datenmodell, d.h., es gibt keine bevorzugte Zugriffs- oder Auswertungsrichtung

### • Anfragebeispiele

Q1: Finde alle Studenten aus Fachbereich 5, die ihr Studium vor 2000 begonnen haben.

```
SELECT *
FROM STUDENT
WHERE FBNR = 'FB5' AND STUDBEG < '1.1.00'
```

Q2: Finde alle Studenten des Fachbereichs 5, die im Fach Datenverwaltung eine Note 2 oder besser erhalten haben.

```
SELECT *
FROM STUDENT
WHERE FBNR = 'FB5' AND MATNR IN
      (SELECT MATNR
       FROM PRÜFUNG
       WHERE FACH = 'DV' AND NOTE ≤ '2')
```

Q3: Finde die Durchschnittsnoten der DV-Prüfungen für alle Fachbereiche mit mehr als 1000 Studenten.

```
SELECT S.FBNR, AVG (P.NOTE)
FROM PRÜFUNG P, STUDENT S
WHERE P.FACH = 'DV' AND P.MATNR = S.MATNR
GROUP BY S.FBNR
HAVING (SELECT COUNT(*)
        FROM STUDENT T
        WHERE T.FBNR = S.FBNR) > 1000
```

## Bewertung – Relationenmodell<sup>2</sup>

### • Informationen des Benutzers

- ausschließlich durch den Inhalt der Daten
- keine physischen Verbindungen, keine bedeutungsvolle Ordnung

### • Deskriptive Sprachen

- hohes Auswahlvermögen und Mengenorientierung
- leichte Erlernbarkeit auch für den DV-Laien

### • Vorteile

- strenge theoretische Grundlage
- einfache Informationsdarstellung durch Tabellen, keine Bindung an Zugriffspfade oder Speichertechnologie, keine Aussage über die Realisierung
- hoher Grad an Datenunabhängigkeit
- symmetrisches Datenmodell; d.h., es gibt keine bevorzugte Zugriffs- oder Auswertungsrichtung
- Parallelisierung möglich, Verteilung der Daten über Prädikate

### • Nachteile

- zu starke Beschränkung der Modellierungsmöglichkeiten
- schwerfällige und unnatürliche Modellierung bei komplexeren Objekten
- Einsatz von nicht-prozeduralen Sprachen „soll“ Ineffizienz implizieren!?
- aber: Optimierung der Anforderungen liegt in der Systemverantwortung

---

2. What's The Greatest Software Ever Written? By Charles Babcock, InformationWeek, Aug. 14, 2006

My No. 2 choice is IBM's System R, a research project at the company's Almaden Research Lab in San Jose, Calif., that gave rise to the relational database. In the 1970s, Edgar Codd looked at the math of set theory and conceived a way to apply it to data storage and retrieval. Sets are related elements that together make up an abstract whole. The set of colors blue, white, and red, for example, are related elements that together make up the colors of the French flag. A relational database, using set theory, can keep elements related without storing them in a separate and clearly labeled bin. It also can find all the elements of a set on an impromptu basis while knowing only one unique identifier about the set.

System R and all that flowed from it—DB2, Oracle, Microsoft SQL Server, Sybase, PostgreSQL, MySQL, and others—will have an impact that we're still just beginning to feel. Relational databases can both store data sets about customers and search other sets of data to find how particular customers shop. The data is entered into the database as it's acquired; the database finds relationships hidden in the data. The relational database and its SQL access language let us do something the human mind has found almost impossible: locate a broad set of related data without remembering much about its content, where it's stored, or how it's related. All that's needed is one piece of information, a primary key that allows access to the set. I like System R for its incredible smoothness of operation, its scalability, and its overwhelming usefulness to those who deal with masses of data. It's software with a rare air of mathematical truth about it.

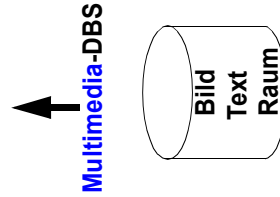
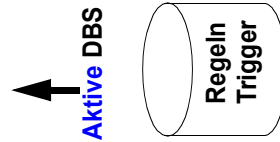
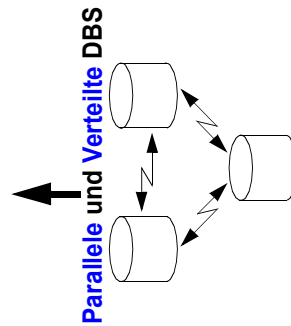
## The Big Picture

WWW-basiertes  
Verarbeitungsmodell    Transaktions-  
verarbeitung    Client/Server-  
Verarbeitungsmodell    ...    Objektorientiertes  
Verarbeitungsmodell

Middleware und Componentware

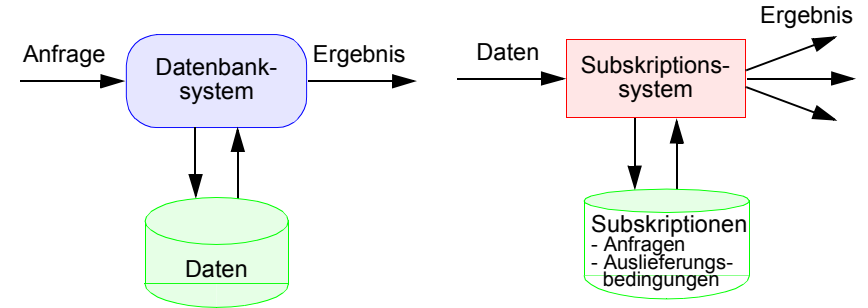
```

SELECT Unfall.Fahrer, Unfall.Vers-Nummer
FROM Unfall, Autobahn
WHERE CONTAINS(Unfall.Bericht, "Schaden"
IN SAME SENTENCE AS
("schwer" AND "vordere" AND "Stoßstange"))
AND FARBE(Unfall.Foto, 'rot') > 0.6
AND ABSTAND(Unfall.Ort, Autobahn.Ausfahrt) < miles (0.5)
AND Autobahn.Nummer = A8;
    
```



## Ein weiteres Paradigma – „Alles fließt (panta rhei)“<sup>3</sup>

### • Vertauschte Rollen



- statt Auswertung von gespeicherten Daten Filterung, Verknüpfung und Transformation von Datenströmen
- zentrale Bedeutung für die individuelle Informationsversorgung, insbesondere bei einer immer weiter fortschreitenden Verwendung vieler kleiner und damit mobiler Endgeräte

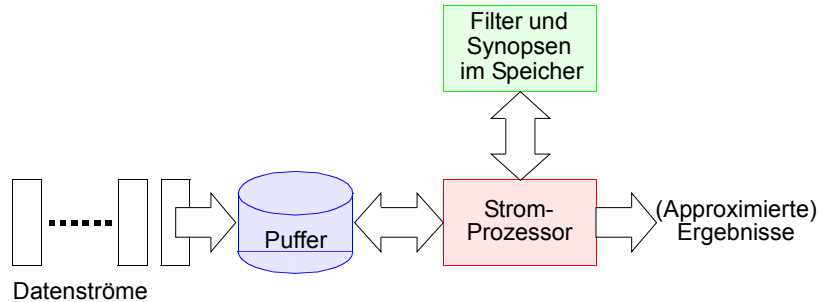
### • Wichtige Unterschiede

Eigenschaft	Datenbankbasierte Informationssysteme	Subskriptionssysteme
Verarbeitungscharakteristik	zustandsorientiert (globaler Zustand)	konsumorientiert (evtl. lokaler temporärer Zustand)
Anfragesemantik	isolierte Anfrage	stehende Anfragen ('standing query')
Zugriffscharakteristik	systemzentriert ('row-set-model')	dokumentenzentriert ('document-model')
Auswertungssemantik	komplexe Analysen	informativ, Auslöser detaillierter Analysen
Schemaaspekt	Existenz eines globalen Schemas	Zugriff auf lokale Schemata der partizipierenden Datenquellen

3. fälschlicherweise Heraklit zugeschriebene Formel für sein Weltbild

## Datenströme – Auswertungsmodell

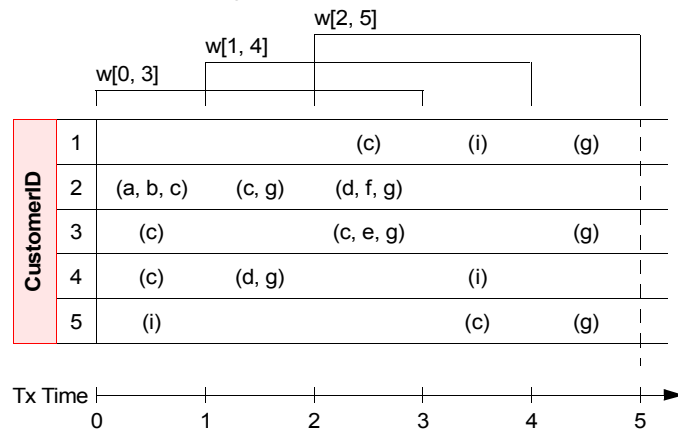
- Daten strömen und sind nicht in einer DB gespeichert<sup>4</sup>



- Datenströme
- Analyse von flachen Datenströmen (z. B. sensorgeneriert)
  - strombasierte Analyse von XML-Dokumenten
  - Wie funktionieren Selektionen und Joins?

- Beispiel für einen Online-Transaktionsfluß

- Warenkorb-AW: Kunden kaufen Waren, die als Ströme zu analysieren sind
- Bestimmung der Häufigkeit von Mustern (frequent pattern identification) ist offensichtlich sehr schwierig!



4. Aktuelles Forschungsthema im Rahmen der dynamischen Informationsfusion

## Verlässliche adaptive DBMS

- DBMS ist nur eine (wichtige) Komponente in einem Informationssystem!

- Höherer Grad an „Selbst-Bewusstsein“ gefordert

- Adaptivität hinsichtlich
  - Benutzer und Arbeitslasten
  - Betriebsmittel, Plattformen und Umgebungen
  - Information (Repräsentation, Inhalt, ...)

- Verhaltensmodelle im DBMS erforderlich

- in den Schichten
- schichtenübergreifend – zusätzliche Kanäle für nicht-lokale Information

- Adaptivität zwischen den Komponenten des Informationssystems

- heterogen und organisationsübergreifend
- Agreement-Protokolle

- Entwicklungsziele (J. Gray: 1998 Turing Lecture)<sup>5</sup>

1. **Trouble-free systems:**

Build a system used by millions of people each day and yet administered and managed by a single part-time person.

2. **Secure system:**

Assure that the system of problem 1 only services authorized users, service cannot be denied by unauthorized users, and information cannot be stolen (and prove it).

3. **Always up:**

Assure that the system is unavailable for less than one second per hundred years – 8 9's of availability (and prove it).

5. J. Gray is the recipient of the 1998 A. M. Turing Award. These problems, strongly related to database systems, are extracted from the text of the talk J. Gray gave in receipt of that award. <http://www.research.microsoft.com/~gray/>