

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.lgis.informatik.uni-kl.de/cms/index.php?id=30>

Datenbankanwendung

Wintersemester 2009/2010

TU Kaiserslautern
Fachbereich Informatik
Postfach 3049
67653 Kaiserslautern

Vorlesung:

Beginn: 28.10.2009

Ort: 48–210

**Zeit: Mi., 11.45–13.15 Uhr
und**

Ort: 46–280

Zeit: Fr., 11.45–13.15 Uhr

Übung:

Ort: 13–222

Zeit: Do., 15.30–17.00 Uhr

Ziele

- **Vermittlung von Grundlagen- und Methodenwissen¹ 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
- Wichtige 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

- 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, 7. Auflage, Oldenbourg-Verlag, 2009
- Kemper, A., Wimmer, M.:* Übungsbuch Datenbanksysteme, 2. Auflage, Oldenbourg-Verlag, 2009
- Kifer, M., Bernstein, A., Lewis, P. M.:* Database Systems – An Application-Oriented Approach, 2nd Edition, Pearson International Edition, 2006
- Saake, G., Sattler, K.-U., Heuer, A.:* Datenbanken – Konzepte und Sprachen, 3. Auflage, mitp, 2008
- Vossen, G.:* Datenmodelle, Datenbanksprachen und Datenbankmanagementsysteme, 5. Auflage, Oldenbourg, München, 2008

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

Notwendigkeit effizienter DBMS

- **Spektrum der Datenarten²**

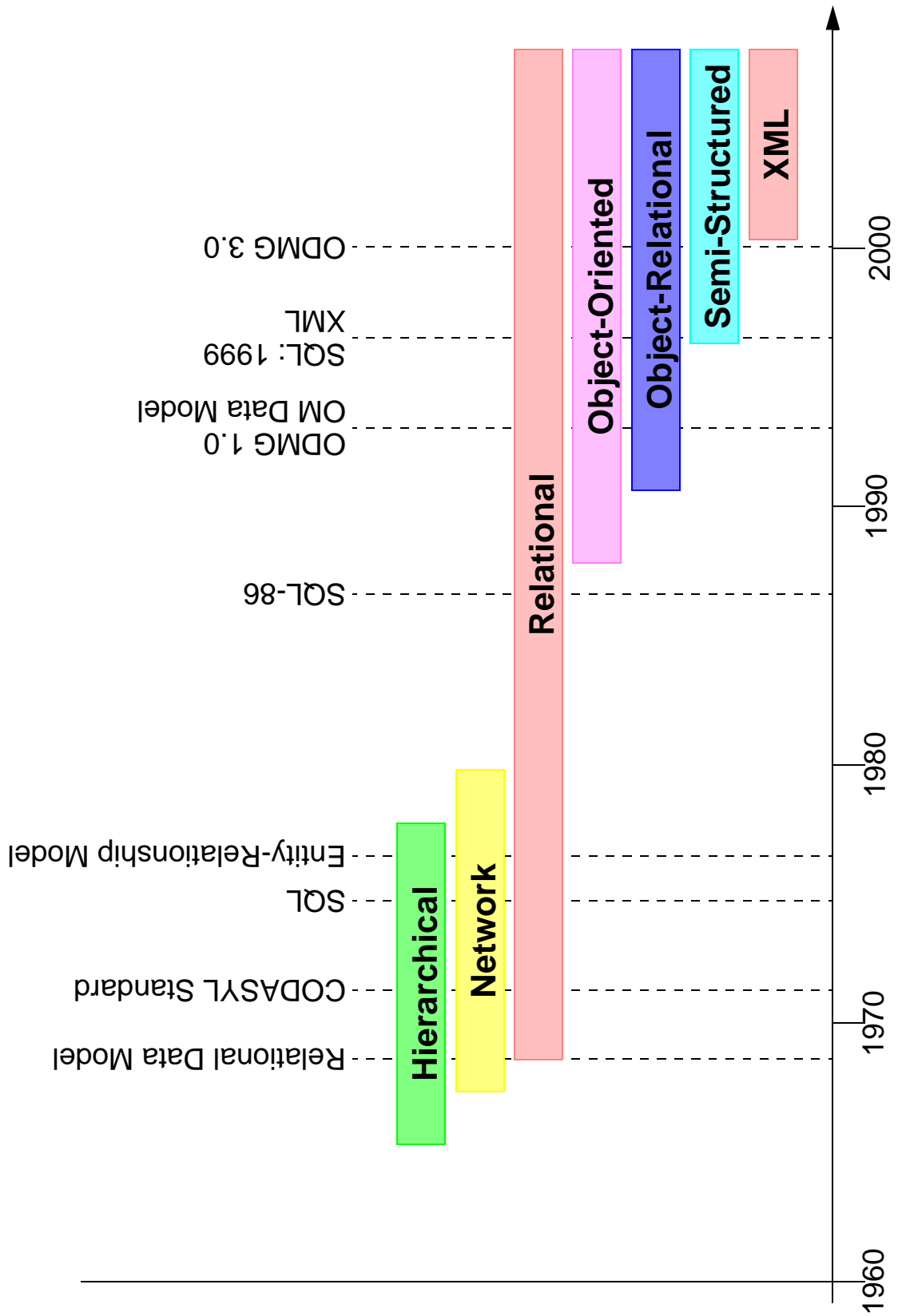
- nicht nur relationale Tabellen, sondern auch VITA-Daten³
- Speicherung, Verwaltung, inhaltsorientierte Suche, Verknüpfung, ...



- „Informationen sind in unserer vom Wettbewerb geprägten Welt ähnlich wie die Luft, die wir atmen – überall vorhanden und absolut lebenswichtig.“
- „Datenbanktechnik ist eine nützliche Infrastruktur wie fließendes Wasser, das wir erst bemerken, wenn es fehlt.“

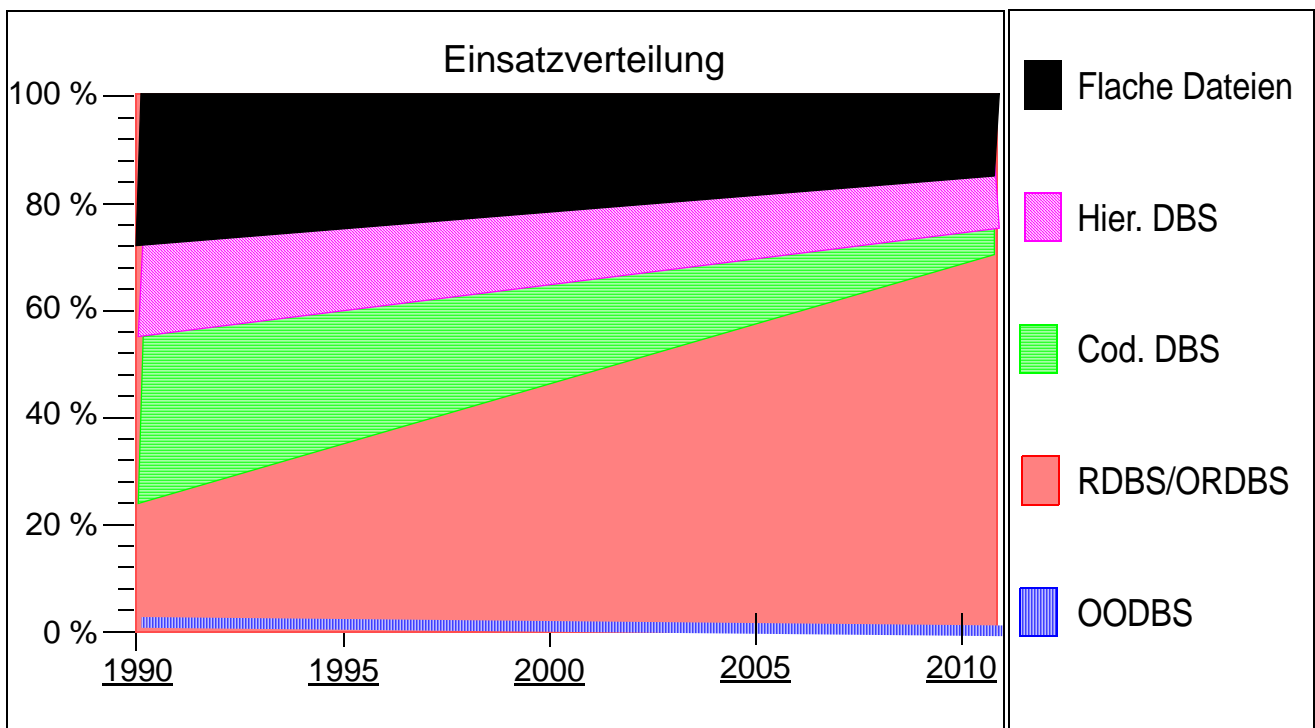
2. Bezeichnungen für Zehnerpotenzen: 3 kilo, 6 mega, 9 giga, 12 tera, 15 peta 18 exa, 21 zetta, 24 yotta und in der anderen Richtung: 24 yecto, 21 zepto, 18 atto, 15 femto, 12 pico, 9 nano, 6 micro, 3 milli
3. How much data is out there? 300 PByte online, 8 EByte offline, 200 EByte analog data (audio, video, etc.) <http://www.sims.berkeley.edu/research/projects/how-much-info-2003/>

Evolution und Historie



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**

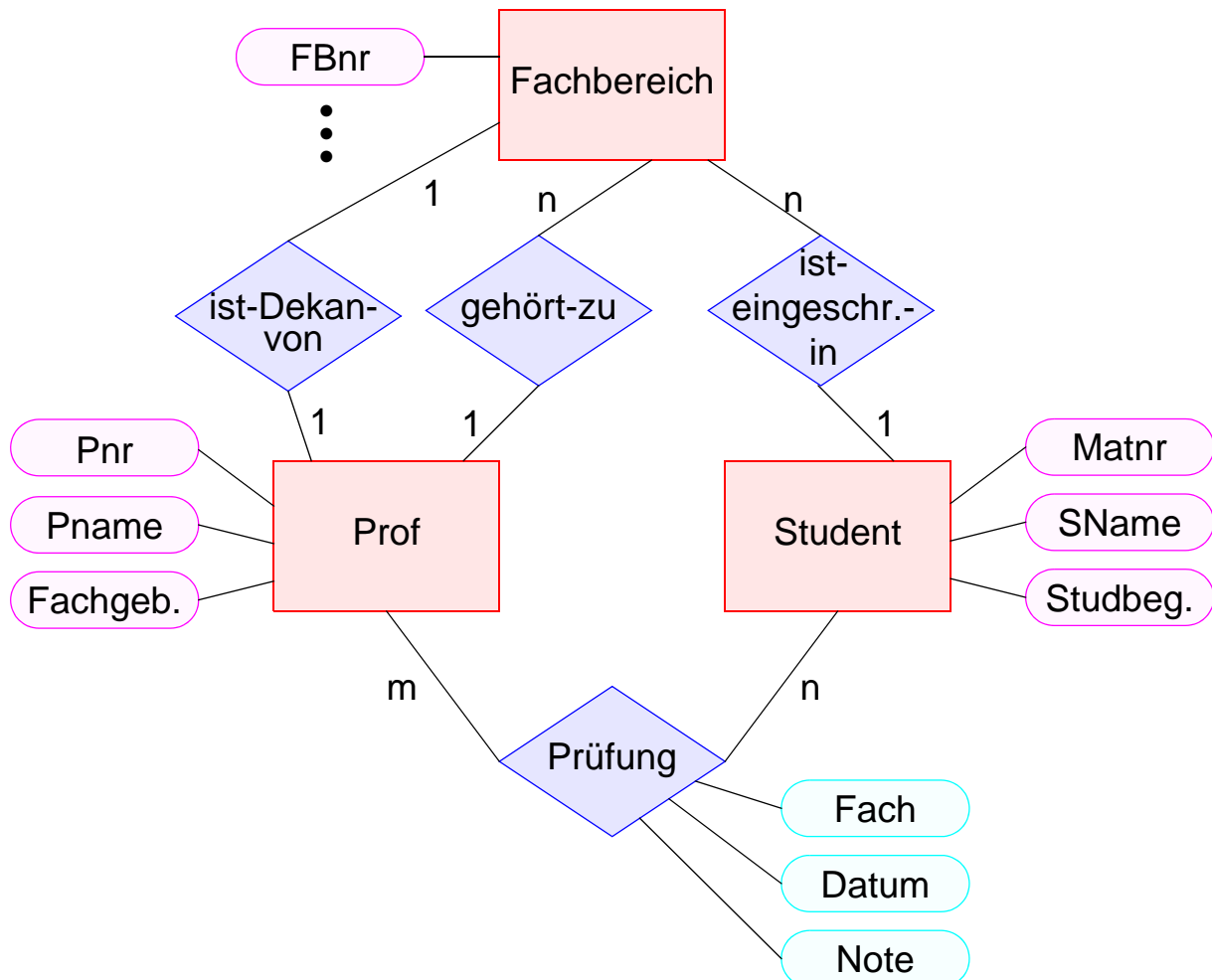
- Aufstellung berücksichtigt nur strukturierte Daten. 85% der weltweit verfügbaren Daten aber sind semi- oder unstrukturiert (Internet, wiss. Aufzeichnungen und Experimente usw.)
- SQL-XML-DBS, XML-SQL-DBS, native XML-DBS, pureXML DBS

- **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	1.10.05
	225 332	MÜLLER	FB 5	15.04.02
	654 711	ABEL	FB 5	15.10.04
	226 302	SCHULZE	FB 9	1.10.05
	196 481	MAIER	FB 5	23.10.05
	130 680	SCHMID	FB 9	1.04.07

PRÜFUNG	<u>PNR</u>	<u>MATNR</u>	FACH	PDATUM	NOTE
	5678	123 766	BWL	22.10.07	4
	4711	123 766	OR	16.01.08	3
	1234	654 711	DV	17.04.08	2
	1234	123 766	DV	17.04.07	4
	6780	654 711	SP	19.09.07	2
	1234	196 481	DV	15.10.07	1
	6780	196 481	BS	23.10.07	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 2005 begonnen haben.

```
SELECT *
FROM STUDENT
WHERE FBNR = 'FB5' AND STUDBEG < '1.1.05'
```

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 – Relationenmodell⁴

- **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

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

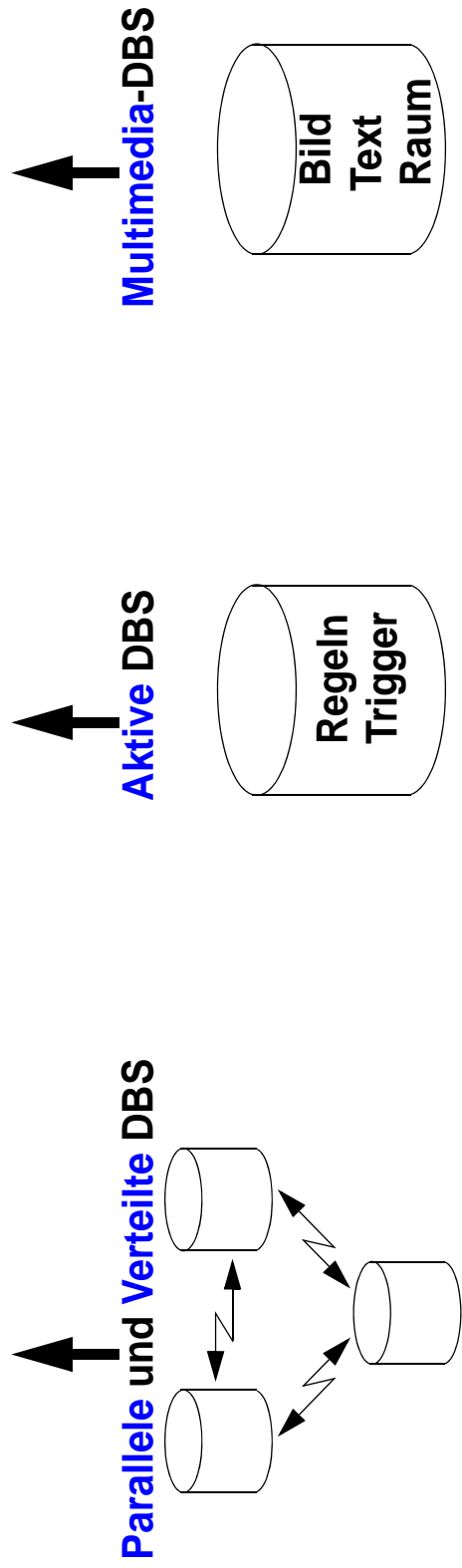
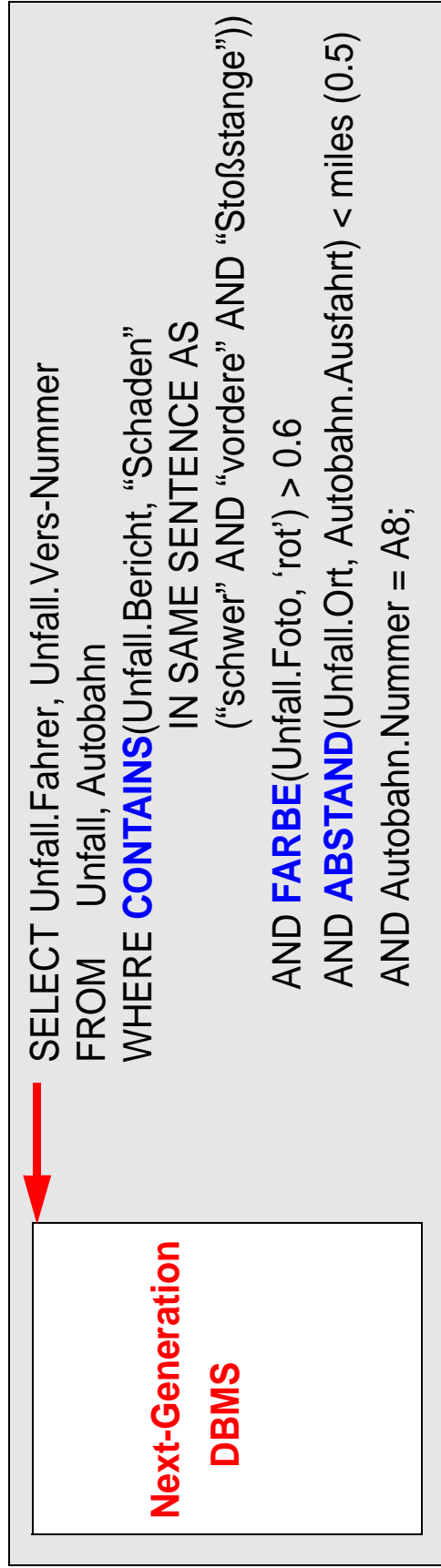
My No. 2 choice is IBM's System R, a research project at the company's Almaden Research Lab in San Jose, Calif., that gave rise to the relational database. In the 1970s, Edgar Codd looked at the math of set theory and conceived a way to apply it to data storage and retrieval. Sets are related elements 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.

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

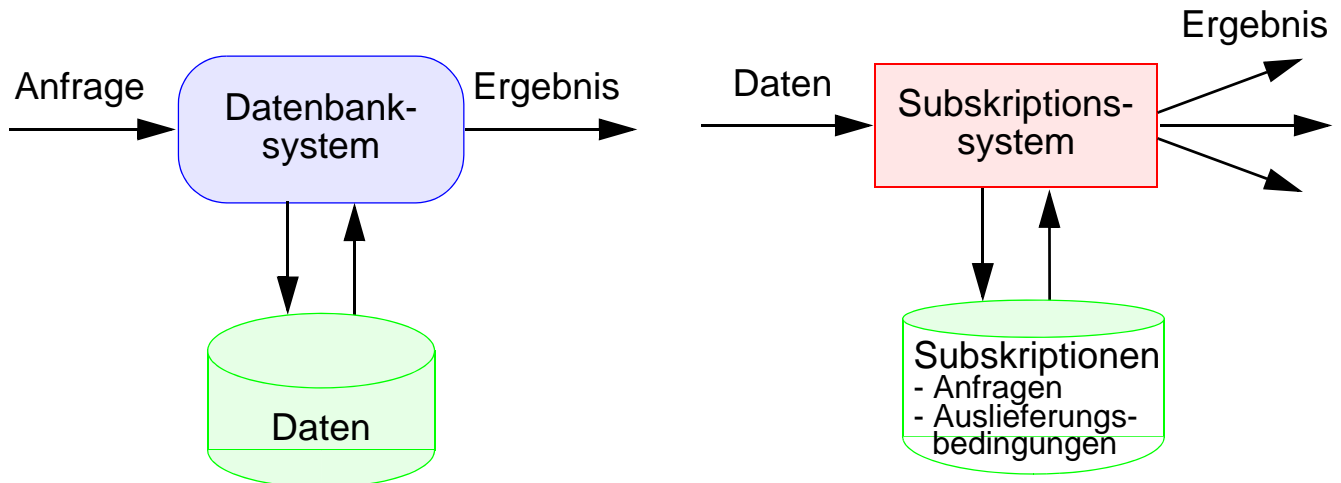
Middleware und Componentware

The Big Picture



Ein weiteres Paradigma – „Alles fließt (panta rhei)⁵“

• 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

5. fälschlicherweise Heraklit zugeschriebene Formel für sein Weltbild

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)⁶**
 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:**

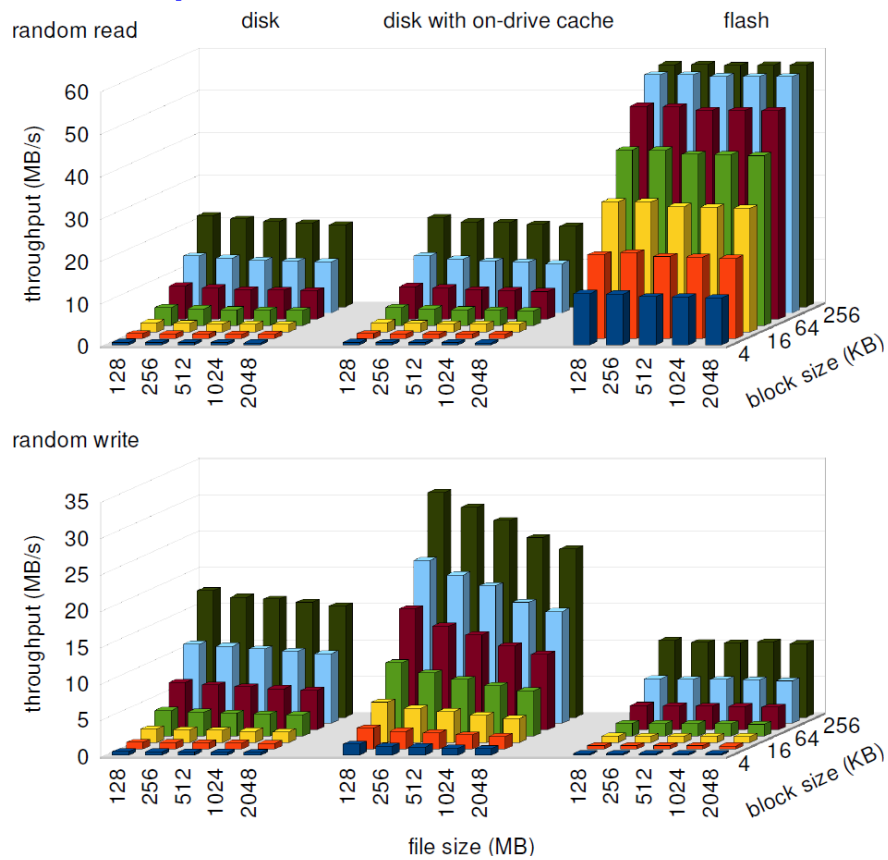
Assure that the system of problem 1 only services authorized users, service cannot be denied by unauthorized users, and information cannot be stolen (and prove it).
 3. **Always up:**

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

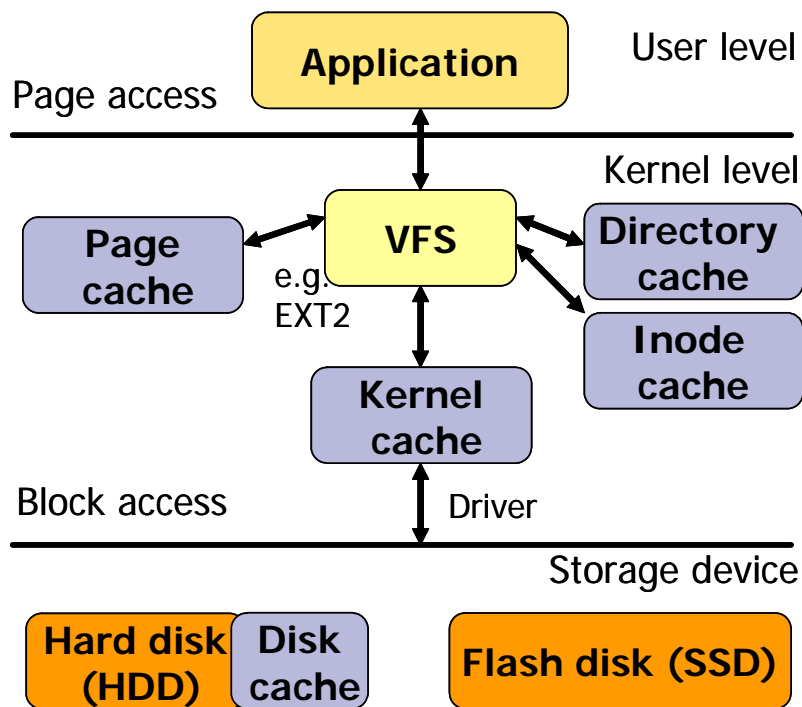
6. J. Gray is the recipient of the 1998 A. M. Turing Award. These problems, strongly related to database systems, are extracted from the text of the talk J. Gray gave in receipt of that award.
<http://www.research.microsoft.com/~gray/>

Energieeffizienz – Es müssen gigantische Datenmengen gespeichert und verarbeitet werden

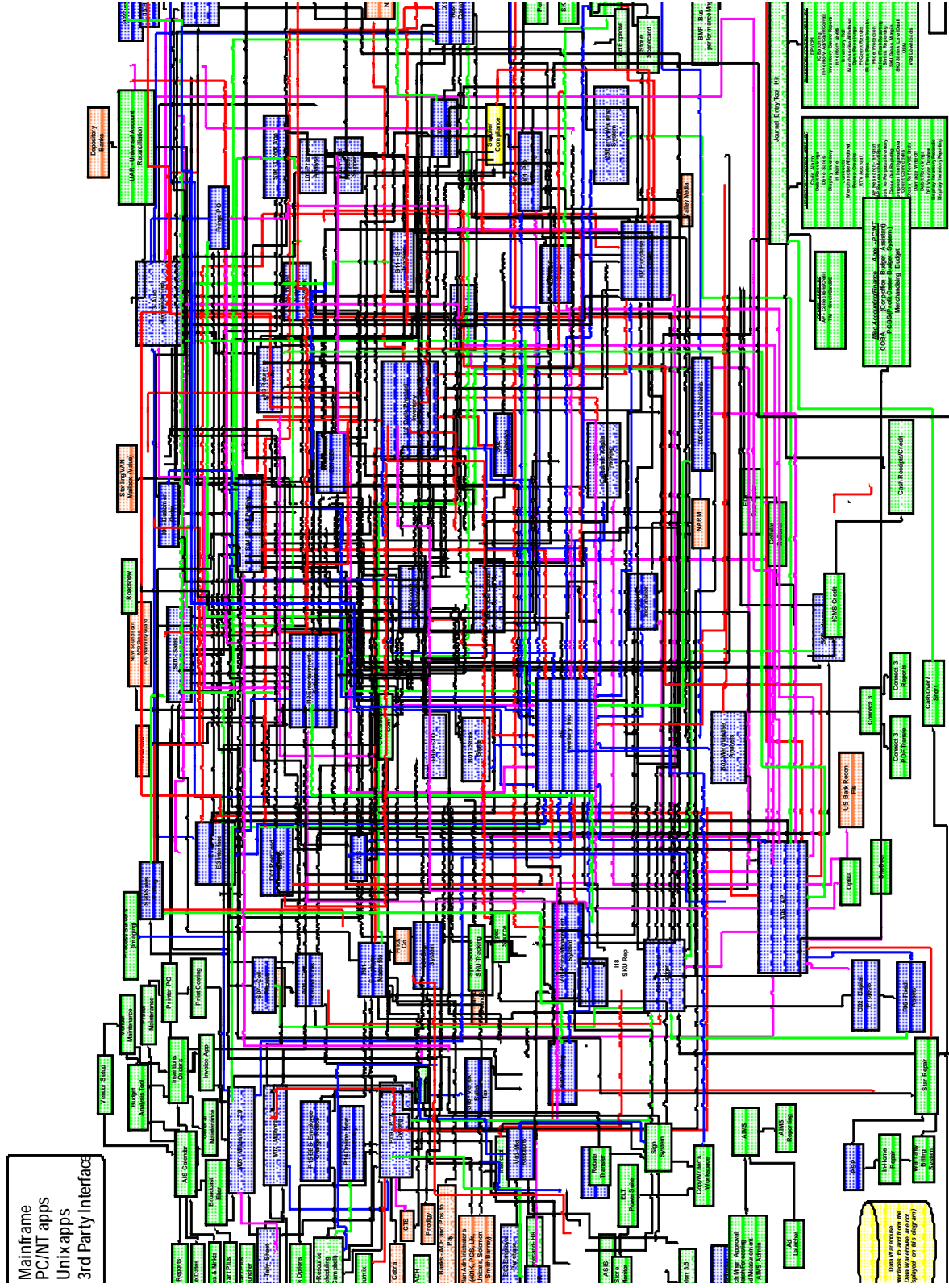
- Einsatz von Flash-Speicher in DBMS



- Wie wirkt sich die Caching-Hierarchie aus?

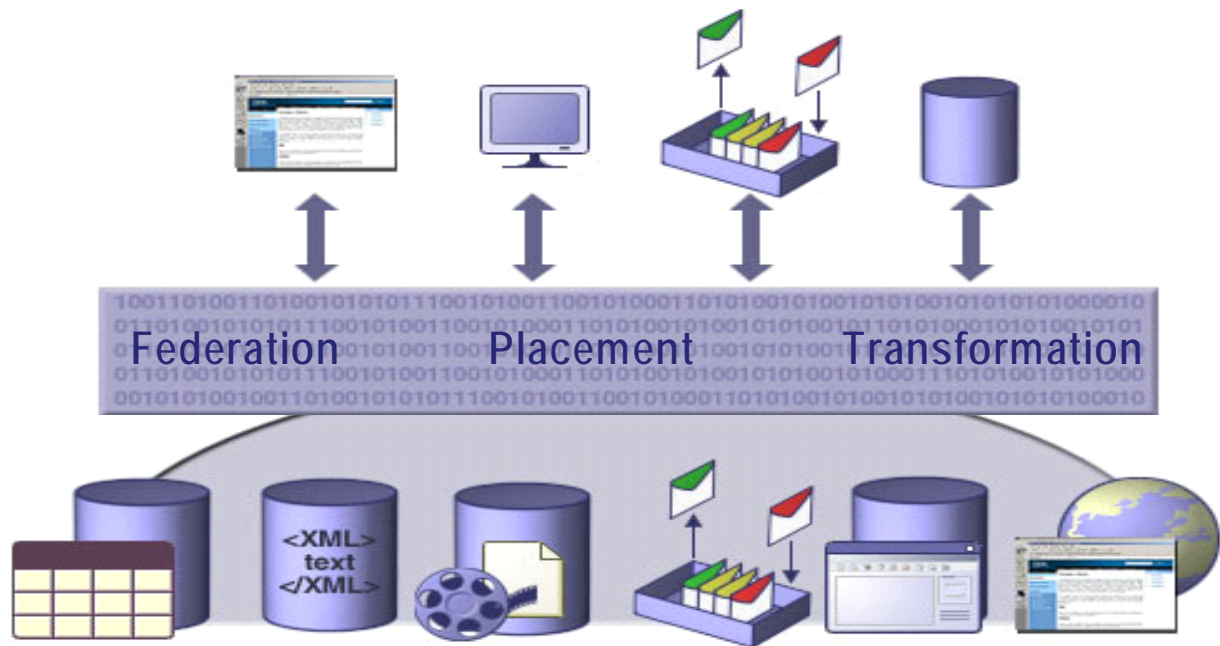


Alltägliches Problem – IT-Infrastruktur



Es gibt keine monolithischen Informationssysteme!

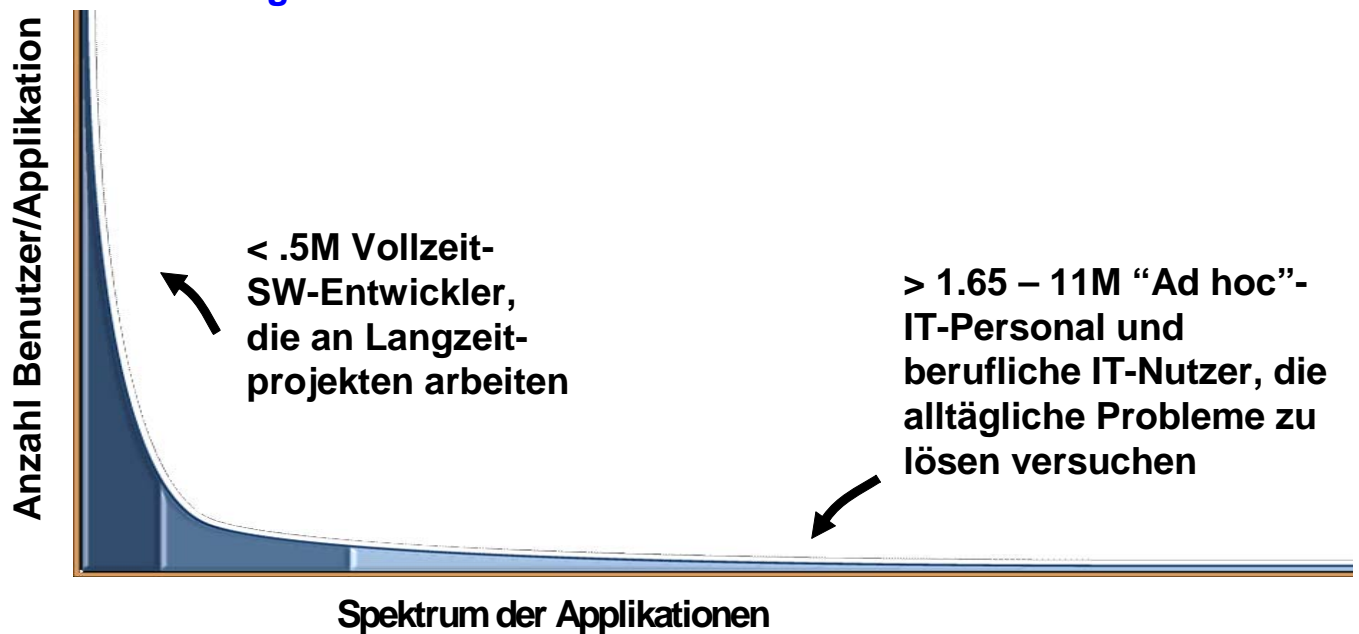
Dynamische Informationsfusion



- **Paradigmenwechsel: bedarfsgetriebene, skalierbare Kopplung und Integration von Datenquellen, -strömen und -analysemodellen (DSMs)**
- **DSMs**
 - persistent gespeicherte Daten
 - kontinuierliche Datenströme, in Zeitfenstern zu verarbeiten (Sensordaten, Web-Nutzerverhalten)
 - abgeleitete DA-Modelle (data mining)
- **Skalierbarkeit**
 - weltweiter Zugriff auf DSMs: Internet, Grid Computing, P2P, Web Services
 - Effiziente Auswahl der geeignetsten Komponenten aus einer großen und schnell wachsenden Menge
- **Dynamik**
 - dynamische Strukturen zur Kooperation in neuen Applikationen: Katastrophen-Management, E-*, ...
 - Ad-hoc-Integration von relevanten DSMs (variable Anzahl, heterogen, autonom, mobil, wechselnde Verfügbarkeit, ...)
- **Datenqualität**
 - Daten variieren in Genauigkeit, Vollständigkeit, Aktualität und Dimensionalität
 - zusätzliche Methoden und Metadaten

Mashups: Neue Form der Integration

- US-Schätzungen für 2006

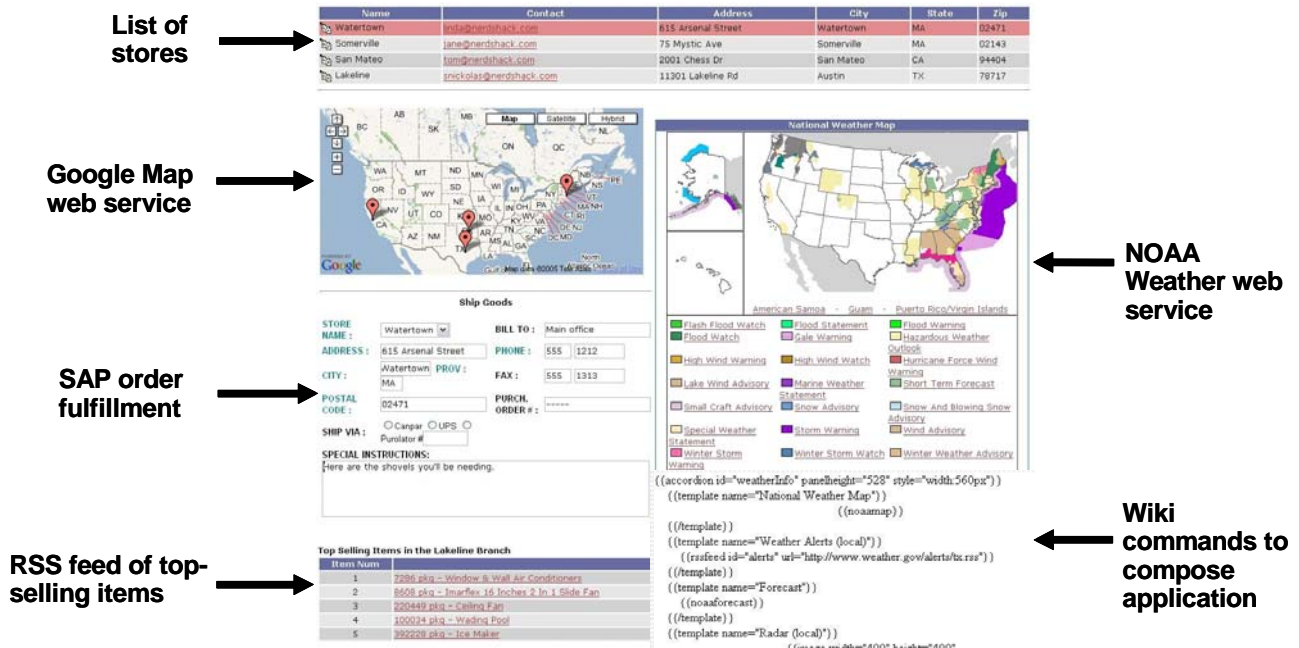


- Welche Eigenschaften haben diese Applikationen?
- Wunschziel: Erstellung durch den Endbenutzer in „5 Minuten“
 - Es dreht sich vor allem um “Information”
 - RSS/Atom, Wikis, Blogs
 - Information Centric Mashups
 - Es entstehen für Web 2.0 “Instant”-Applikationen
- Web 1.0: seit 1996, Einwegkommunikation, klassische Websites
- Web 1.5:
DotCom-Hochzeit, dynamische Websites, SOA: Flexibilität, Reuse, Shopsysteme und CMS. Web 1.5 erreichte die Trennung von „Information“ and „Prozess“
- Web 2.0
ermöglicht im Web-Kontext dieselbe Trennung von “Daten” und “Logik”,
 - die in den 80er Jahren die Nutzung von Datenbanken revolutioniert hat
 - Web 2.0 außerhalb und innerhalb eines Unternehmens
 - wird nur mit einer „Info 2.0 Mashup Fabric“ erfolgreich sein
 - Annahme: Entwicklung von Instant-Applikationen auf Basis von Mashups wird der bedeutendste Treiber für Web 2.0 in Unternehmen werden

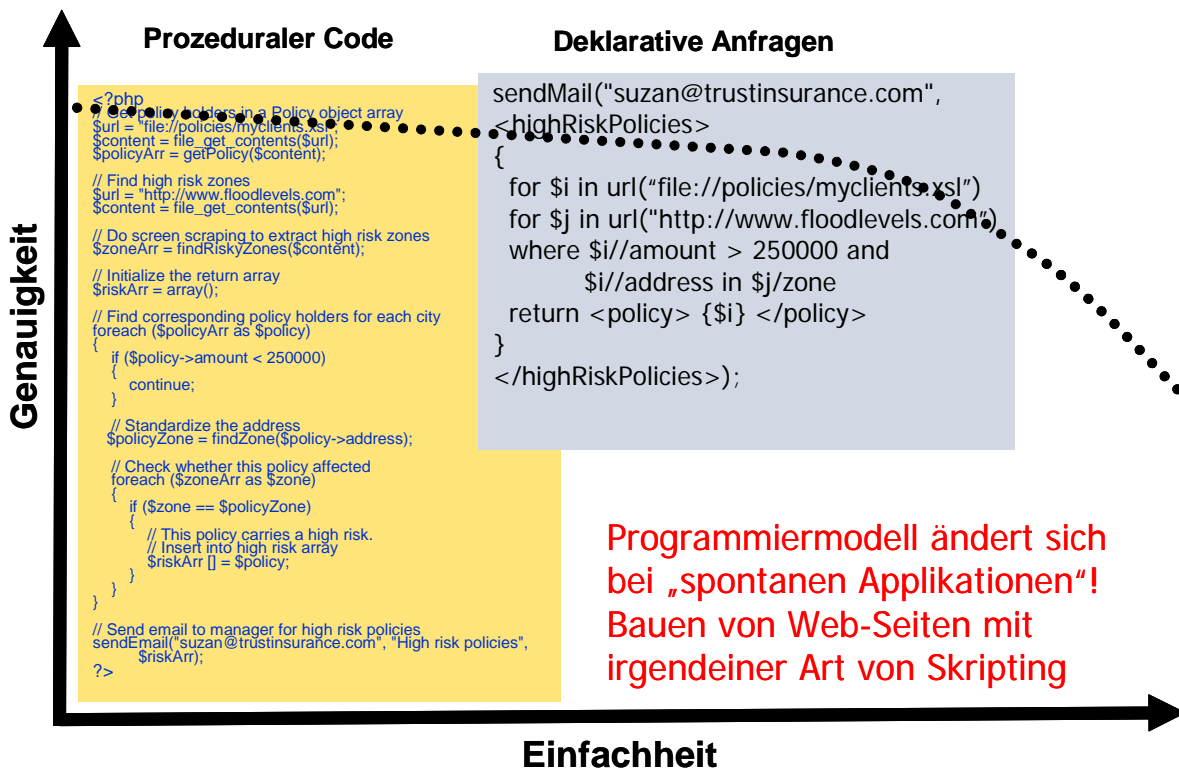
Wie können Mashups einfach erstellt werden?

• Der Mashup (Vermischung)

bezeichnet die Erstellung neuer Inhalte durch die nahtlose (Re-)Kombination bereits bestehender Inhalte. Der Begriff stammt aus der Welt der Musik und bedeutet dort soviel wie Remix. In den deutschen Sprachraum wurde er rund um das Schlagwort Web 2.0 importiert, da Mashups als ein wesentliches Beispiel für das Neue an Web 2.0 angeführt werden: Inhalte des Webs, wie Text, Daten, Bilder, Töne oder Videos, werden z.B. collageartig neu kombiniert.



• Skripting und deklarative Anfragen



MyLifeBits

- **Selbst wenn der Mensch sein Leben lang kein Wort vergessen würde, ist die Speicherung der Informationsmenge heute schon leicht machbar⁷**

Bilanz (durchschnittliche Zahlen für US-Bürger/Jahr)

	Stunden	Wort/Minute	Wörter/Jahr	MBytes
TV	1578	120	11 Mio.	50
Film	12	120	-	-
Lesen	354	300	6,4 Mio.	32

? ~6 GBytes von ASCII-Daten (Text) in 75 Jahren

Die automatische Erfassung dieser Informationen (Spracherkennung, und OCR oder elektronische Bücher/Zeitungen und ASCII-Skripte von TV-Sendungen) ist mit tragbaren Geräten möglich.

? Das gilt nicht für Ton, Bild und Bewegtbild!

- **Aber in 20 Jahren?**

Speicherbedarf für eine Person (nach Jim Gray)

Datentyp	Datenrate (Bytes/ Sek.)	benötigter Speicher pro Stunde und Tag	benötigter Speicher für eine Lebenszeit
gelesener Text	50	200 KB; 2-10 MB	60-300 GBytes
gesprach.Text @120 wpm	12	43 K; 0,5 MB	15 GBytes
Sprache (komprimiert)	1.000	3,6 MB; 40 MB	1,2 TBytes
Bewegtbild (komprimiert)	500.000	2 GB; 20 GB	1 PByte

? Aufzeichnung der gesamten Lebensgeschichte wird möglich

- **Weltweites Wachstum**

- Professionelle Content Provider (Journalisten usw.): 2 GB Text pro Tag
- Benutzergenerierte Web-Inhalte: dramatisch (siehe social networks)

7. <http://research.microsoft.com/barc/MediaPresence/MyLifeBits.aspx>; <http://en.wikipedia.org/wiki/MyLifeBits>

Schlussfolgerungen

- **Es wird genug Platten- und Bandspeicher geben, um alles zu speichern, was alle Menschen schreiben, sagen, tun oder photographieren.**
 - Für das Schreiben gilt dies bereits heute
 - In einigen Jahren trifft das auch für die restlichen Informationen zu
 - Wie lange wird es noch dauern, bis alle VITA-Dokumente eines Menschen als Lebensgeschichte aufgezeichnet werden können?
 - **Rechner speichern und verwalten Informationen besser und effektiver als Menschen**
 - Viele Platten und Kommunikationsverbindungen speichern direkt Informationen aus **Rechner-zu-Rechner-** und nicht mehr (nur) aus Mensch-zu-Mensch-Kommunikation
 - Wie lange wird es noch dauern, bis der Mensch die meiste gespeicherte Information gar nicht mehr zu sehen bekommt?
 - Wir müssen lernen, wie alles automatisch ausgewertet werden kann und was bei unserer knappen Zeit unserer besonderen Aufmerksamkeit bedarf.
 - **Künftige Entwicklung**
 - Heute konzentriert man sich bei den „Digitalen Bibliotheken“ auf die Eingabe: auf das Scanning, Komprimieren und OCR von Informationen.
 - Morgen wird anstelle der Eingabe die „relevante Auswahl“ die wesentliche Rolle spielen: Selektion, Suche und Qualitätsbewertung von Informationen
- ? **Wir können eine reale „World Encyclopedia“ mit einem echten „planetary memory for all mankind“ aufbauen, wie H.G. Wells bereits 1938 in seinem Buch „World Brain“ geschrieben hat!**