Datenstrukturen
und
Datenorganisation
unter besonderer Berücksichtigung von Datenmodellierung

Prof. Rudolf Bayer, Ph.D.
www3.in.tum.de
TU München, WS 2001/02

Inhaltsverzeichnis

Kap. 1 Einleitung
Kap. 1.1 Architektur von IV-Systemen

Mehrschichten-Ansatz:

Anwendungen

Konzetuelle Modelle

Logische Modelle

Physische Modellierung = Implementierung

Problem:

Geeignete Abbildung über alle Abstraktionsebenen?

3

Randbedingungen

- Anwendungen: t/s, DB-Größen, updates/s,
  Responsezeit, Archiv-Größe,
  Zuverlässigkeit (z.B. 24x7),
- Annahmen zur Ableitung solcher Kennzahlen?
- verfügbare Basistechnologien
- Kosten, Fertigstellungsdatum,

4

Beispiele für Anwendungen

Telekom Rechnungen	human genome
OMNIS/Elektra	GfK
Bank	Patentamt
Bayer. Staatsbibliothek	VW-Zulieferung JIT
VW-Reparaturen	
Amadeus	
Krankenversicherung AOK	

5

Beispiel Telekom Rechnungen

20 Mio Anschlüsse x 500 Anrufe/Monat

= 10^{10} Anrufe/Monat

DB Größe

100 B/Anruf ~ 10^{12} B/Monat = 1 TB/Monat

Transaktionsrate

10^{10} Anrufe/Monat = $3x10^{8}$ Anrufe/Tag ~ $3x10^{3}$ Anrufe/sec

Tagsüber ~ $6x10^{3}$ Anrufe/sec

insert into Anrufe

(Tel#, Datum, Start, Ende, Tel2#, Einheiten, Provider)

6

Telefon Rechnungen erstellen

select Tel#, provider, sum(Einheiten)

from Anrufe

group by Tel#, provider

Sortiere Anrufe nach (Tel#, provider, Datum, Start)

Frage: Zeit, um 1 TB Daten zu sortieren?

Alternative Architekturen zu einem zentralen System?

7

Beispiele für konzeptuelle Modelle

- E/R
- ME/R
- Objekt orientierte Modelle (oo)
- Html
- Xml
- UML
- Prozeß-Algebra
- Petri Netze
- Dexter Modell
- Amsterdam Modell

8

Beispiele für logische Modelle

- Hierarchische (IMS)
- Netzwerke (UDS)
- Relationale DBMS ~ 10 Mrd $ Weltmarkt
- Deduktive DBMS (1980-1995), LOLA, NAIL!, ECRC, MCC
- Objektorientierte DBMS (Object Store, Versant, Ontos, Objectivity) ~ 100 Mio $ Weltmarkt ~ 1%
- NF^2 (non firsr normal form), ähnlich zu XML

9

Beispiele für Basistechnologien

- Prozessoren + Multiprozessoren
- Speicher + Caches
- Busse (intern und extern)
- Festplatten und RAIDS
- Betriebssysteme und Prozesse, Threads
- Netze und Protokolle (Netzzugriff vs Plattenzugriff)
- Architekturen (Client-Server)

10

Kap.1.2 Anwendungsbeispiel: OMNIS/Myriad

Ziel: Literatur-Verwaltung,
multimediale Dokumenten-Verwaltung

Konzeptuelles Modell der Bibliotheken
Hierarchien von Dokumenten
Zeitschrift
Band
Heft
Aufsatz
Monographie

11

Maschinelles Austauschformat Bibliotheken MAB

Autoren*
Titel
Untertitel
Verleger
Jahr
Ort
Zeitschrift
Band*
Heft*
erste Seite
letzte Seite
Fingerprint ...

Ca. 950 einzelne Felder !!!

12

Konzeptuelles Modell von OMNIS/Elektra

- Vollständiger Text
- Dokument als Folge von Seitenbildern
- Treffer mit Kurzinformation

Methoden

- Recherchieren, Freitext
- Kurzinfo anzeigen
- Dokument bestellen, drucken, lesen, lokal speichern
- Bezahlen
- Zitieren

Hinweis: Modell geht weiter als heutige Realität in klassischen Bibliotheken, Methoden werden von unterschiedlichen Akteuren veranlaßt

13

Recherche: z.B. Freitext-Suche in 5 GB (konzetuelle Ebene)

- exakte Einzelbegriffe
- Pattern-Match mit Wildcards:
  datenbank%
  dat%ba%
- Phrasensuche: relational algebra
  query opti%
- Boolsche Kombinationen
  dat%ba% & dat%stru%
  datenba% & optimier%
- Nachbarschaftssuche
  hierarchy .... space
- Struktursuche: Attributkombinationen
  ?Autor = ´Bayer´ & ?Jahr ≤ 1986
  & ?Jahr > 1978
  & dat%ba% & synchroni_ation

14

Dokumenten DB, logisches Modell von OMNIS

Dok = (Text, DB-Satz, Image-Folge, Postscript-Datei)

noch zu ergänzen: Details des DB-Satzes, z.B. Dublin Core, MAB, UniMarc, ...

Text Repräsentation für Freitext Recherche:
My0 (Wort, W#) **My1** (W#, Doc#, Pos)

typische DB-Größe: 10^6 Dokumente
$5 \cdot 10^6$ Images x 40 KB = $2 \cdot 10^{11}$ Bytes = 200 GB
Text: $5 \cdot 10^6 \cdot 10^3$ B = 5 GB
Speicherung dieser DB?

15

Das physische Modell:

Erfassungsvorgang

1. **Einscannen** → Pixelbild = Image

   • geeignete Datenstruktur? z.B. Spezialform von Quad-Tree
   • Kompressionsalg., 1 x auf Server
   • Dekompression: < 2sec auf PC 486
   • Speicherung in DB
   • Verschickung über Netz

16

2. Text erzeugen: OCR

OCR (Image) → Text fehlerhaft!!

- Abgleich 1 Seite Text ≈ 200 Wörter
  mit Stammreduktion
  Wörterbuch: 10^5 - 10^6 Einträge
- Korrekturvorschläge
  für ca. 10 Wörter
- Darstellung der Textseite?
- Darstellung des Wörterbuchs für Abgleich
- Darstellung des Wörterbuchs für Korrektur

17

3. Archivierung

- Indexierung des Textes
- Erzeugung von Strukturdaten
- Index warten
- Dok auf mehrere DBen verteilen

Gefragt:
- Datenstrukturen für alle Be- und Verarbeitungsvorgänge!
- Spezielle Anforderungen der Vorgänge berücksichtigen!

18

Arbeitsschritte/Zeitabschnitte	1.	2.	3.	4.	5.	6.
1. Mensch: Dokument einlegen Scan starten	D2			D3		
2. Rechner: scannen und speichern		D2			D3	
3. Mensch: Bereich wählen OCR starten			D2			D3
4. Rechner: OCR ausführen	D1			D2		
5. Mensch: korrigieren Maske füllen Archivierung starten		D1			D2	
6. Rechner: archivieren			D1			D2

19

Retrieval auf logischer Ebene

datenba%

select My1.Doc# **from** My0, My1

where My0.Wort **like** datenba% **and** My0.W# = My1.W#

hierarchy ... space

My01	**My11**		**My12**	**My02**
W#	Doc#		W#	
	Pos	<	Pos	
	Pos+6	>	Pos	

20

```
select My11.Doc# from
        My0 My01, My1 My11, My1 My12, My0 My02
        where My01.Wort   =    ‚hierarchy‘   and
              My01.W#     =    My11.W#      and
              My02.Wort   =    ‚space‘       and
              My02.W#     =    My12.W#      and
              My11.Doc#   =    My12.Doc#    and
              My11.Pos    <    My12.Pos     and
              My11.Pos+6  >    My12.Pos
```

Physische Ebene: Datenstrukturen, schnelle Bearbeitung solcher Queries?

21

Kap. 1.3 Vorlesungsziel

Zwei zentrale Anliegen

- Anwendung modellieren
- Geeignete Datenstrukturen, um Komplexität mehrerer Operationen **gleichzeitig** niedrig zu halten:
  - Zeitkomplexität
  - Platzkomplexität

mit Randbedingungen von Anwendungen!

z.B. insert, update, delete, find, compress, decompress, bulk load, ... retrieve interval, sort (z.B. Sätze variabler bzw. fester Länge)

22

Def.: O(...) - Notation, ***Ordnung einer Funktion***:

Seinen $\boldsymbol{N_o}$, $\boldsymbol{R_o}$ nicht-neg. integer, real

Sei X entweder $\boldsymbol{N_o}$ oder $\boldsymbol{R_o}$,

$g : X \to X$

$O(g)_{def} := \{f : x \to x \mid \exists\, c > o,$

$\exists x_o \in X,\ \forall x \geq x_o : f(x) \leq c \cdot g(x)\}$

Notationen:

$f \in O(g)$, $f = O(g)$, f hat Ordnung O(g), f ist von Ordnung O(g)

Eigenschaften:

$f_1 \in O(f) \wedge g_1 \in O(g) \Rightarrow f_1 \bullet g_1 \in O(f \bullet g)$

$f \in O(g) \wedge g \in O(h) \Rightarrow f \in O(h)$

$f \in O(g) \Rightarrow O(f) \subseteq O(g)$

23