

Multimedia-Datenbanken

Kapitel 1: Einführung und Übersicht

Friedrich-Alexander-Universität Erlangen-Nürnberg
Technische Fakultät, Institut für Informatik
Lehrstuhl für Informatik 6 (Datenbanksysteme)

Prof. Dr. Klaus Meyer-Wegener

Wintersemester 2002 / 2003

Technische Universität Kaiserslautern
Fachbereich Informatik
AG Datenbanken und Informationssysteme

Dr. Ulrich Marder

Wintersemester 2003 / 2004

Multimedia-Systeme

- **Multimedia in/auf/mit Rechnern:**
 - Erfassen und Präsentieren von Information in **unterschiedlichen Medien**
 - (**Medium** \cong Informationsträger)
- **Eingabe:**
 - keine Umsetzung in ein anderes Medium (kein "Abtippen" oder Erfassen)
- **Ausgabe:**
 - kann das geeignetste Medium benutzen (ggf. Umsetzung durch das System)

Multimedia-Systeme (2)

- **einerseits Erhöhung der "Informations-Bandbreite"**
 - bessere **Benutzerschnittstellen**

- **aber auch mehr Information im System:**
 - Inhalt eines Bildes (einer Tonaufnahme, ...)
niemals vollständig
in Medien wie Text oder Graphik wiederzugeben

Multimedia-Anwendungsklassen

- **Archivierung**
 - Presse
 - Büro
 - Krankenhaus
 - Bibliotheken, Museen

- **Unterrichtung / Werbung / Unterhaltung / Assistenz**
 - Schule, Universität
 - Weiterbildung
 - Fernsehen
 - Spiele
 - Lexika

Multimedia-Anwendungsklassen (2)

- ❑ **Schreiben / Publikation / Gestaltung / Entwurf**
 - Presse
 - Ingenieurwesen
 - Architektur
- ❑ **Überwachung**
 - Fabriken
 - Verkehr
 - Wetter
 - Militär

Zur Erinnerung: Datenbanken – wann?

bei einer großen Menge von Daten

auch, aber vor allem:

- ❑ **wohlstrukturiert**
- ❑ **redundanzfrei**
- ❑ **flexibel abfragbar (recherchierbar)**
- ❑ **vielseitig verwendbar (offen für neue Anwendungen)**
- ❑ **von mehreren Anwendungen gleichzeitig nutzbar,
bei hoher Aktualität der Daten**
- ❑ **ausfallsicher**

Datenbanken – womit?

- **Datenmodell und Schema**
 - Datenmodellierung (DB-Design) schwierig und aufwändig
 - enge Kooperation zwischen Entwickler und Anwender
- **Normalisierung**
- **Anfragesprache (query language)**
 - Selektionen, Verknüpfungen, Aggregationen
- **Synchronisation der Zugriffe**
- **Transaktionen**
 - "Alles oder nichts" für eine Sequenz von Änderungen
 - automatische Reparatur von inkonsistenten Zuständen

Multimedia-Datenbanksysteme

- **was kann Datenbanktechnik für Multimedia leisten?**
 - **Datenunabhängigkeit** der Anwendungen
 - Anwendungsneutralität der Datenstrukturen (Schema)
 - Unterstützung der Suche
 - Speicherorganisation
- **Basisdienst**
 - für die Vielzahl der Anwendungen ("Infrastruktur")
 - nicht so sehr ein Endbenutzersystem, Programmschnittstelle!
- **Speichern und Wiedergewinnen**
 - von (Multi-) Media-Datenobjekten

Multimedia-DBS (2)

- ❑ **Geräte- und Formatunabhängigkeit**
 - beim Zugriff auf MM-Datenobjekte
- ❑ **Beziehungen**
 - von MM-Datenobjekten untereinander
 - wie auch zwischen MM-Datenobjekten und strukturierten Daten
 - darstellenund zum Auffinden benutzen:
 - Unterstützung der **Navigation**
- ❑ **Inhaltsorientierte Suche**
 - deskriptiver Zugriff
 - unscharf (Ähnlichkeit)

Multimedia-DBS (3)

und natürlich:

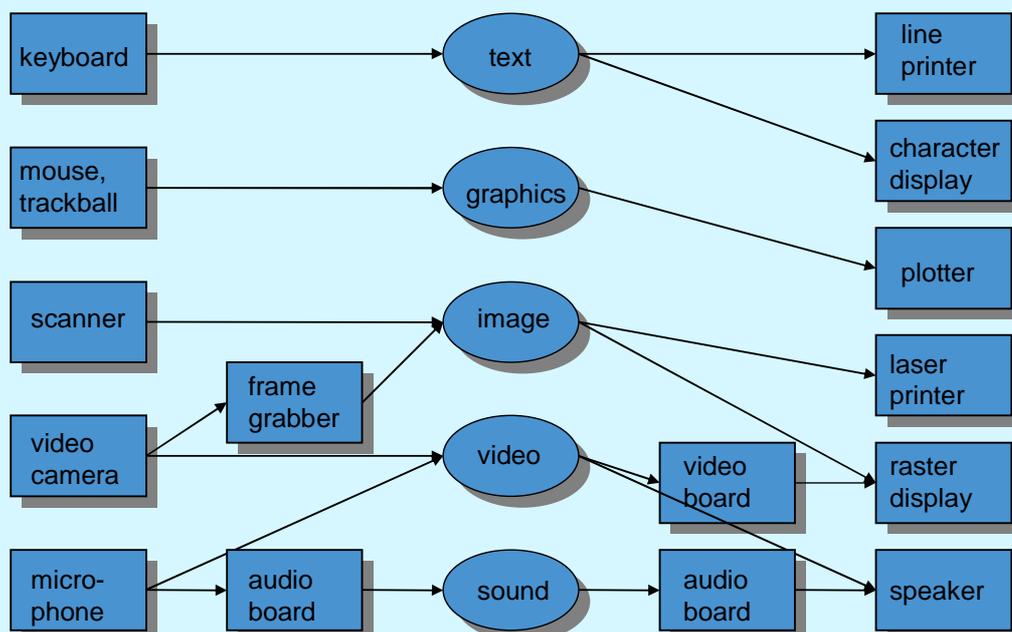
- ❑ **Wiederherstellung im Fehlerfall**
- ❑ **Mehrbenutzerbetrieb, Synchronisation**
- ❑ **Zugangskontrolle**
- ❑

Was ist neu?

- ❑ **Suche nach Ähnlichkeit**
 - nicht nur exakte Übereinstimmung
- ❑ **sehr spezielle Zugriffspfade**
 - oft multidimensional
 - evtl. auch Graphen
- ❑ **Speicherverwaltung**
 - große Objekte
 - Einbeziehung von Abspielzeiten
- ❑ **Auslieferung der Daten**
 - zeitgesteuert
 - dauert signifikante Zeit

Multimedia-Daten

- ❑ **durch Ein-/Ausgabegeräte bestimmt:**



Multimedia-Daten (2)

□ im Vergleich:

Medium	Elemente	Anordnung	typische Größe	zeit-abhängig?	Sinn
Text	druckbare Zeichen	Folge	10 KB (5 S.)	nein	visuell /akustisch
Graphik	Vektoren, Flächen, Körper (3D), Texturen	Menge oder Folge (Z-Ordnung)	10-100 KB	nein	visuell
Rasterbild	Bildpunkte (Pixel)	Matrix	1 MB (1024X1024)	nein	visuell
Tonaufnahme	Lautstärke-pegel	Folge	600 MB (Audio-CD)	ja	akustisch
Bewegtbild (Video-Clip)	Rasterbild, Graphik	Folge (Script)	2 GB (30 min.)	ja	visuell

Bezeichnungen

- **Medienobjekt**
 - (oder Medien-Datenobjekt)
 - ein Datenobjekt, das einem *einzigem* Medium angehört, also ein einzelnes Bild oder ein Textstück
- **Multimedia-Objekt**
 - (Multimedia-Datenobjekt, auch "Mixed-Mode Object")
 - Aggregation (Komposition) von Medienobjekten unterschiedlichen Typs, z.B. Video (Bild + Ton)
- **Multimedia-Daten**
 - Sammelbegriff für Medienobjekte und Multimedia-Objekte
- **Multimedia-Dokument**
 - aggregiert Medienobjekte und Multimedia-Objekte
 - legt räumliches und ggf. zeitliches Layout fest
 - kann zusätzlich Strukturen für Navigation/Browsing besitzen (z. B. Links)

Unformatierte Daten

wichtige Unterscheidung:

□ formatierte (strukturierte) Daten

(NAME = "Müller"; GEBDAT = "520623",)

- maximale Länge (= endlicher Wertevorrat)
- Werte von Variablen, Feldern, Attributen; durch Namen beschrieben
- Bedeutung weitgehend vorgegeben
- relativ geringer Informationsgehalt
- (klassische Datenbank-Technik)

Unformatierte Daten (2)

□ unformatierte (unstrukturierte) Daten

"Er heißt Müller. Geboren ist er am 23. Juni des Jahres 1952. "

- beliebige Länge
- teilweise selbstbeschreibend
- Bedeutung nur schwach vorgegeben
- hoher Informationsgehalt
- (Information Retrieval)

Medienobjekte

- ❑ aus formatierten *und* unformatierten Daten zusammengesetzt
- ❑ Rohdaten
 - unformatiert (s. oben)
 - lange Folge (Menge, ...) von kleinen Elementen
(*Bits, Buchstaben, Pixel, Linien, Energieniveaus, ...*)
- ❑ Registrierungsdaten (Steuerungsdaten)
 - obligatorisch
 - erforderlich für korrekte **Interpretation** und **Identifikation** der Rohdaten
 - Interpretation: welche Struktur? was bedeuten die Elemente?
 - Identifikation: Unterscheidung ansonsten gleicher Objekte
(z. B. Zeitpunkt der Aufnahme, aufgenommenes Objekt, ...)

Medienobjekte (2)

- ❑ Beschreibungsdaten
 - optional
 - oft redundant:
Darstellung der **Struktur** und/oder des **Inhalts**
in einem anderen Medium
 - formatiert oder unformatiert (auch kombiniert)

Operationen auf Medienobjekten

- ❑ **Erzeugen (eingeben, erfassen, capture)**
 - mit Gerät – aus Programm – aus Datei
- ❑ **Ausgeben (zeigen, präsentieren, present)**
 - auf Gerät – an Programm – in Datei
- ❑ **Bearbeiten (modifizieren, editieren)**
- ❑ **Zusammensetzen**
 - Erzeugen *multi*-medialer Objekte
- ❑ **Weitergeben (versenden)**

Operationen auf Medienobjekten (2)

- ❑ **Archivieren**
- ❑ **Auswerten (aggregieren, ableiten)**
 - Filtern, Analysieren,
Erzeugen von Beschreibungsdaten
- ❑ **Suchen (vergleichen)**
 - Mustererkennung auf Rohdaten
 - Ähnlichkeit
 - oder inhaltsorientiert auf Beschreibungsdaten

Daten zum Medienobjekt "Rasterbild"

□ Rohdaten

- Matrix von Bildpunkten
(Pixel = Picture Element, auch "Pel" genannt)

□ Registrierungsdaten

- Anzahl Bits pro Pixel ("Pixeltiefe", typisch: 1, 8, 24)
- Anzahl Pixel pro Zeile (Breite des Bildes)
- Anzahl der Zeilen (Höhe des Bildes)
- Art der linearen Abspeicherung: zeilenweise oder spaltenweise
- Bedeutung eines Pixels: Grauwert, Farbdefinition, Index einer Farbtabelle (Palette, Colormap)

Daten zum Medienobjekt "Rasterbild" (2)

□ Registrierungsdaten (Forts.)

- ggf. Farbtabelle mit bestimmter Anzahl von Einträgen und Länge dieser Einträge (meist 24 Bit)
- ggf. Art der Farbdefinition: RGB, IHS, YIQ,
- und anderes mehr

□ Beschreibungsdaten

- Text, Schlüsselworte, Wissensrepräsentation
- Graphik: erkannte Linien und Flächen
- darauf aufbauend zweidimensionale Objekte wie Kreise, Rechtecke usw.

Operationen auf Rasterbildern

- **Erzeugen (eingeben, erfassen, capture)**
 - mit Gerät: Scanner, Kamera
 - aus Programm: Hauptspeicherstrukturen für Rasterbilder (*Ximage, Pixrect, ...*)
 - aus Bild-Datei (Sun Rasterfile, GIF, TIFF, GEM-Image, PBM, ...)
- **Ausgeben (zeigen, präsentieren, present)**
 - auf Gerät: Bildschirm, Drucker
 - an Programm: in Hauptspeicherstruktur
 - in Datei: Wahl eines bestimmten Formats
- **Bearbeiten (modifizieren, editieren)**
 - pixel-orientierte Zeichensysteme ("Paintbrush-Systeme"): *MacPaint, PhotoShop, ...*
 - Setzen einzelner Pixel
Ändern der Farbtabelle
Bitmap-Operationen: Überlagern anderer Bilder
 - oft unerwünscht aus dokumentarischen Gründen

Operationen auf Rasterbildern (2)

- **Weitergeben (versenden)**
 - Nachrichtenformate (Austauschformate) für Bilder: *MIME, IIF*
- **Archivieren**
 - Bild-Datenbanken (Pictorial Databases)
- **Auswerten (aggregieren, ableiten)**
 - Kontrastverstärkung, Vergrößern oder Verkleinern (Zoom), Umsetzung Farbe nach Grauwert und Ausschnittbildung (window) – erzeugen wieder Bilder
 - Linienerkennung – erzeugt Graphik
 - Optische Zeichenerkennung (OCR) – erzeugt Text

Operationen auf Rasterbildern (3)

□ Suchen (vergleichen)

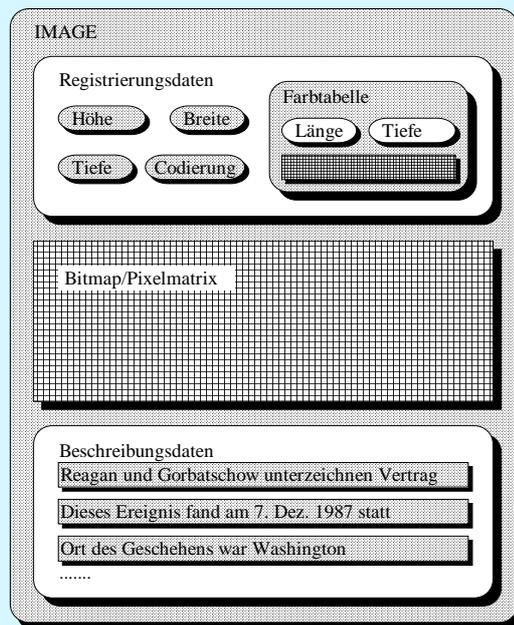
- Mustererkennung (Pattern Matching):
 - wird unterstützt durch Zugriffspfade (Iconic Indexing)
 - wichtig für einige Bildanalysetechniken, allgemein aber nicht sehr aussagekräftig
 - Anwendungsbeispiele für Mustererkennung:
 - Materialprüfung (Produktfehler erkennen)
 - Satellitenbilder (Bodenschätze, archäologische Artefakte etc. erkennen)
 - In MM-DBS: Merkmalsextraktion (Formen, Texturen, ...)
- Bild-Bild-Vergleich:
 - z.B. Phantombild oder Photo mit Archivbild zur Identifikation von Personen
 - Ähnlichkeitsmaße? – schwierig

Multimedia-Datenbank-Technik

- **Einführung der (elementaren) Datentypen**
 - TEXT, GRAPHIC, IMAGE, SOUND, VIDEO,
mit darauf anwendbaren Funktionen (→ Abstrakte Datentypen)
- **Einbettung in existierende Datenmodelle**
 - Relationenmodell (als Domains)
 - objektorientiertes Modell (als Klassen)
- **Nutzung der verfügbaren Modellierungskonstrukte:**
 - Relationen bzw. Klassen
 - Attribute bzw. Instanzvariablen
 - Primärschlüssel bzw. Objektidentifikator
 - Methoden, Vererbung, ...
- **und der Abfragesprachen:**
 - Relationenalgebra, SQL

Beispiel: der Datentyp IMAGE

- abstrakte Sicht auf eine Instanz:



Beispiel: der Datentyp IMAGE (2)

- **Operationen**
 - beschreiben den Datentyp vollständig
 - Funktionen, d.h. liefern Ergebnis bestimmten Typs
- **Zugriff: Ausgeben**

lesender Zugriff auf Teile:

```
interface Image {  
    public int height ( );  
    public int width ( );  
    ...
```

ganzes Bild in ein Programm (als SUN-Pixrect):

```
public Pixrect getPixrect ( );
```

auf ein Gerät:

```
public boolean display (Device d);
```

Beispiel: der Datentyp IMAGE (3)

□ Auswerten, ableiten

```
public int pixelcount (byte [ ] pixelvalue);
```

zählt die Häufigkeit eines bestimmten Pixelwerts;

```
public Image window (int x0, int y0, int x1, int y1);
```

bildet einen Ausschnitt

□ Modifizieren

Prozeduren (keine Funktionen):

```
public void replacePixelvalue (  
    int x, int y,  
    byte [ ] pixelvalue  
);
```

u.v.a.

Beispiel: der Datentyp IMAGE (4)

□ Erzeugen (eingeben)

a) aus einem Programm:

```
class ImageClass implements Image {  
    public ImageClass (  
        int height,  
        int width,  
        int depth,  
        float aspectRatio,  
        Code encoding,  
        int colormapLength,  
        int colormapDepth,  
        int [ ] [ ] colormap,  
        byte [ ] pixelmatrix  
    );
```

in einem spezifischen Systemkontext (hier: SUN) auch:

```
public ImageClass (Pixrect pr, Colormap cm);
```

b) aus einer Datei:

```
public ImageClass (String filename, String format);
```

c) von einem Gerät:

```
public ImageClass (Device d);
```

Suche (Selektion)

- ❑ **hierarchisch definierte Suchbereiche**
- ❑ **Benutzung zugeordneter Attribute mit Standard-Datentypen:**
 - beschränkter Informationsgehalt
- ❑ **Benutzung der Medienobjekte selbst:**
 - Browsing (MINOS)
 - Pattern Matching
 - Ordnung, Vergleich (POSTGRES)
Gleichheit von Bildern?
 - Klassifikation (Schlüsselworte)
 - Inhaltsrepräsentation
 - Statistik, Merkmale, Features
 - Text (Inhaltsangabe)
 - Semantische Netze
 - Frames
 - Prädikate (Logik)
 -

MMDBS-Architektur

viele offene Fragen:

- ❑ **ein homogenes System (erweiterbar, konfigurierbar, ...)
oder ein Rahmen für eine "Föderation" von medienspezifischen
Systemen?**
- ❑ **interaktive Schnittstelle
und/oder
Programmschnittstelle ?**
- ❑ **Transaktionskonzept**
- ❑ **Verteilung
Multimedia-Workstation und Server**