

Multimedia Content Management (13)

Prof. Ralf Möller, TUHH, STS

■ Previous lectures:

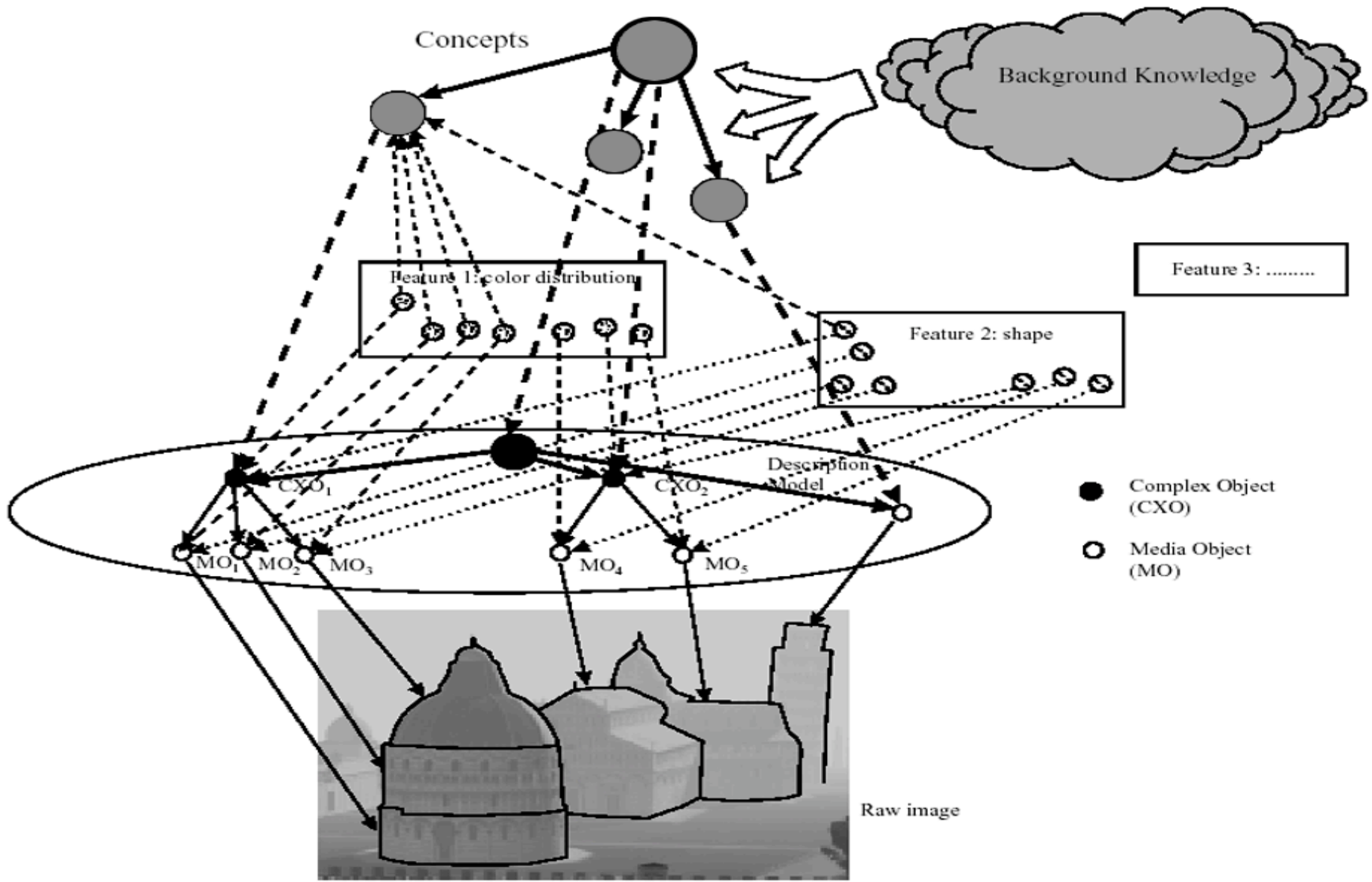
- Description-based retrieval (based on metadata) with respect to axioms restricting the meaning of terms
- Multimedia data structures

■ Today:

- Content-based retrieval based on multimedia databases (for mass data but without axioms, i.e., names are „primitives“)
- Multimedia query languages

■ Inspired by talks from:

- Volker Severt (MOQL) and Niko Zenker (MOQL, MTQL)
 - http://www.witi.cs.uni-magdeburg.de/iti_db/lehre/spezialseminarws0102/



Agenda

- Requirements for a MM query language
- Object Query Language
vs. TIGUKAT Query Language
- Content-based & fuzzy queries
- Spatial Primitives
- Temporal Primitives
- Presentation functions
- Summary

General requirements

- Familiarity of potential users required
- SQL syntax successful / commodity
- Object-orientation ensures orthogonal design

- Apparent idea:
- Extension of an existing query language namely OQL (Object Query Language) for multimedia purposes

Object Query Language (OQL)

- Orthogonal language
- SQL92 + object-oriented extensions (structured objects, object identity, path expressions, polymorphism, methods, late binding, ...)

- Syntax

SELECT [***DISTINCT***] projection_attributes

FROM query [[***AS***] identifier] {, query [[***AS***] identifier]}

[***WHERE*** query]

[***GROUP BY*** partition_attributes]

[***HAVING*** query]

[***ORDER BY*** predicate {, predicate}]

Object Query Language: Example

```
SELECT STRUCT (a:e.age, s:e.salary)
FROM (SELECT f
        FROM Employees f
        WHERE f.seniority>20)
AS e
WHERE e.name = „John“
```

Extensions of OQL (in **WHERE** part)

■ Spatial part

■ Expressions

- spatial objects, lines, points, circles, ...

■ Functions

- Length, area, intersection, ...

■ Predicates

- Media_object **CONTAINS** salientObject

■ Temporal part

■ Expressions

- Intervals, points, ...

■ Functions

- Duration, ...

■ Predicates

- Before, after, ...

Extensions of OQL (in **PRESENT** part)

- Constraints for the **presentation** of objects in the result set of the query
 - AtWindow, AtTime, Display, ...

- -> Multimedia Object Query Language: **MOQL**

TIGUKAT Query Language

- Pronounced: tee-goo-kat, means „object“ in the Inuit language
- Support for typical features of object-oriented languages
- SQL-like (ad-hoc), with path expressions
- Basic structure
 - **SELECT** object_variable_list
 - **[INTO [PERSISTENT [ALL]]]** collection_reference
 - **FROM** range_variable_list
 - **WHERE** boolean_formula

Multimedia TIGUKAT Query Language (MTQL)

- Additional terms in WHERE part
 - Spatial expression, functions, predicates
 - Temporal expression, functions, predicates

- Example:

```
SELECT m  
FROM m IN C_image, p IN C_person  
WHERE m CONTAINS p
```

Content-based and Fuzzy Queries

- Information gained from extracted feature or metadata
 - Keywords
 - Features
 - Concepts
- How to cope with ...
- ... too large results sets?
 - Query relaxation
- ... too small results sets?
 - Query expansion

Spatial Primitives (1)

- Requirements for spatial query languages
 - Hybrid queries
(combining the spatial and the abstract part)
 - Which spatial relations or spatial properties to consider?
 - Deals with composition of complex objects
 - Combination of time and space
 - Spatial index structures
(e.g., k-d trees, quadtrees, rtrees, ...)

Spatial Primitives (2)

- Automatic or semi-automatic derivation of descriptors for images
- Descriptor structure (see MPEG-7 approach)
 - <SalientSet, SpatialSet, ColorSet, TextualSet>
 - SalientSet: homogeneous regions for main objects in image
 - SpatialSet: spatial properties of salient objects
 - minimum bounding rectangles (MBR), positions, center of gravity, ...
 - ColorSet: Color descriptor for salient objects and background
 - TextureSet: Texture of salient objects or background

Spatial Primitives (2)

- Two images can be in the following relations
 - identical
 - Equality in salient, spatial, color, and textual set parts
 - coincident
 - Identical spatial sets
 - contains
 - Part of one picture (salient object) is identical with other picture
 - similar
 - W.r.t. spatial, color, and texture properties

Spatial Primitives (4)

■ Basic structures (2D)

- Points, lines, regions
- Specific forms of regions
 - Circles, rectangles

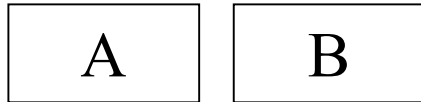
■ Spatial functions

- Distance (euclidian)
- ...

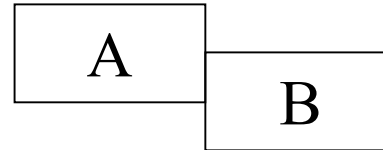
■ Spatial relations (predicates)

- Directions (cardinal directions)
 - left, right, above, below, front, back, south, north, west, east, northwest, ...
- Topological relations

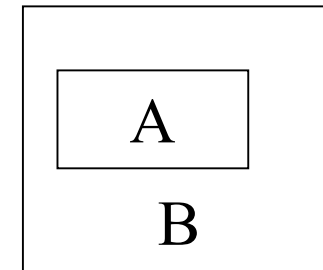
Topological relations



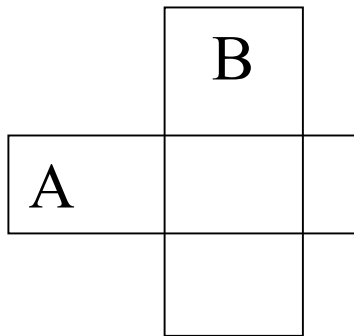
A disjoint B



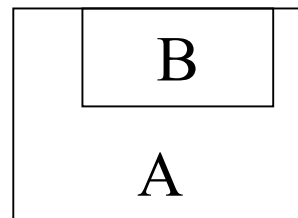
A touch B



A inside B
Inverse: B contains A



A overlap B



A cover B
B covered_by A



A equal B

Spatial functions and operators

■ Functions

- mbr – Minimal bounding rectangle
- position
- distance
- region - conversion
- pan – only parts of an image

■ Operators (for data manipulation language)

- resize – change size
- superimpose – image composition

Spatial Queries

■ MOQL

```
SELECT o.name  
FROM SalientObjects o, Images m  
WHERE m CONTAINS o AND (o INSIDE a  
    OR a COVER o)
```

■ MTQL

```
SELECT o.B_name  
FROM o IN C_salientObject, m IN C_image  
WHERE m CONTAINS o AND (o INSIDE a  
    OR a COVER o)
```

Spatial Primitives for 3D

- Polyhedrons
- Relations: Front, back, front-left, front-right, ...

Temporal Primitives

- Essential for retrieval languages
- In particular for historical databases
- Temporal relations between real-world object/events and relations between multimedia data (salient objects in images)

- Examples for queries:
 - “Find latest video with Gerhard Schröder”
 - “Latest” requires special treatment in query languages

Temporal Primitives

■ Time interval

- 13 interval relations (J.F. Allen 1983)
- equal, before, after, meet, metBy, overlap, overlapedBy, during, include, start, startedBy, finish, finishedBy

■ Time instant

- Anchored point in time
- Relations (<, >)
- Addition, subtraction (new instant)
- Can be seen as time interval with same lower bound and upper bound

■ Time span

- Non-anchored duration
- Independent of interval or instant
- Examples:
 - Year, month, day, hour, minute, second, ms

Temporal Functions

- Unary: LB (lower bound), UB (upper bound), length
- Binary: union, intersection, average, difference

From Images to Media

■ Video data

- Sequence of clips (sequences of frames, i.e., images)
- Each frame associated with a time instant
- Each video/clip associated with time interval
- Same properties as images have
 - Color, mbr, textures,...
- Additionally: temporal relations between parts, e.g., movement

■ Audio data

- Sequence of tones
- Each tone associated with length, frequency and intensity
- Additionally: temporal relations (e.g., in speech)

Prerequisites: Video/Audio Analysis

- Local
 - Object movements
- Global
 - Camera movements
 - Positional (6 degree of freedom)
 - Optical (zoom, focus)
 - Functions: zoomIn, zoomOut, panLeft, panRight, tiltUp, tiltDown, cut, fade, wipe, dissolve
- Clips
 - Must be segmented
 - Can be seen as sets of salient object at different times

Temporal Queries, Example

■ MOQL

- **SELECT** c1, c2
- **FROM** Clips c1, Clips c2, p.prSet pr
- **WHERE** c1 **CONTAINS** p **AND** c2 **CONTAINS** p **AND** pr.timestamp **DURING** intersection(c1.timestamp, c2.timestamp)

■ MTQL

- **SELECT** c1, c2
- **FROM** c1 **IN** C_clip, c2 **IN** B_clip, pr **IN** p.prSet
- **WHERE** c1 **CONTAINS** p **AND** c2 **CONTAINS** p **AND** pr.B_timestamp **DURING** intersection(c1.B_timestamp, c2.B_timestamp)

Presentation Functions

- Synchronization of audio and video data
 - Compare with SMIL
- Spatial layout
 - Number of (sub-)windows
 - (Relative) sizes, positions
- Temporal layout
 - Which object to start first, length of presentation
- Scenarios (spatial + temporal)

Presentation Functions (2)

- `atWindow(identifier, point, point)`
- `atTime(absoluteTime)`
- `display(identifier, start_offset, duration)`
- `play(identifier, start_offset, duration, speed)`
- `thumbnail(identifier)`
- `resize(identifier, width, height)`
- `parStart`
- `parEnd`
- `after`

Presentation Functions, Example

■ MOQL

- **SELECT** m
- **FROM** Images m
- **WHERE** m **CONTAINS** p
- **PRESENT** resize(m, 100, 100) **AND** display(m, 0, 10)

■ MTQL

- **SELECT** m
- **FROM** m IN C_image
- **WHERE** m **CONTAINS** p
- **PRESENT** resize(m, 100, 100) **AND** display(m, 0, 10)

Uncertainty, Imprecision, and Similarity

- MultiMedia Query Language (MMQL)
- BilVideo Query Language
- Uncertainty: Weighting of query parts
- Imprecision: Recognition degree of parts
- Similarity: Matching to a specific degree

MMQL Query Syntax

- Query ::= select <select-stmnt>
 - from <from-stmnt>
 - where <condition>
- select-stmnt ::= [n-hits] <select-list>
- from-stmnt ::= CLASS, DCLASS, **U**, **I**, **X**
- condition ::= einfach vs. komplex

Predicates / Operators

- sim: similarity
- match: degree of recognition

- and: both parts high
- or: one part high
- not: negation

Functions / Operators

- `o.feature(fid)`: feature value of an object
- `v.rec-degree`: degree of recognition
- `mo.part-of`: reference to the container
- `v.attribute`: component of a structured value
- `o.method(args)`: procedure call

Example 1

```
select 10 R.oid  
from REGIONS as R  
where  
    R.feature(color) sim red, 0.7 and  
    R.feature(shape) sim circle, 0.3;
```

Example 2

```
select S.oid
from SHOTS
where
  S.oid in (
    select F.part-of
    from FRAMES as F
    where
      F.oid match CHURCHES)
and S.rec-degree > 0.7;
```

Multimedia databases

- Research prototypes
 - MOQL, MTQL, MMQL
- Commercial products
 - Informix DataBlades
 - IBM DB2 Extenders
 - Oracle *interMedia*

Summary

- Extensions of SQL/OQL
- Primitives
 - Spatial: points, lines, regions
 - Temporal: instants, intervals
- Functions
- Predicates / Relations
 - Quantitative constraints
 - Qualitative relations

- Complex queries can be hard to understand
- Graphical user interfaces (e.g., VisualMOQL)

VisualMOQL

VisualMOQL Interface - Mozilla {Build ID: 2001101117}

File Bearbeiten Ansicht Suchen Gehe Lesezeichen Tasks Hilfe Debug QA

http://darwell.uwaterloo.ca/~ddbms/DISIMA/Interface.html

Documentation

Image Class: City_image

Images Returned: 25

Similarity Threshold (%): 80

Vehicle: [dropdown]

LSO Class	Attributes	Subclasses
Bicycle	type	
Motorcycle	make	
Truck	model	
Car	year	
Plane		
Ship		

Work Canvas : Revise Query

Hospital H101

Bicycle B102

Car C103

Car C104

Insert Delete Clear Edit Move Resize

Insert Relation Delete Relation Image Property Validate

Query Canvas

Query 1 AND NOT Query 2 OR Query 3

Edit Delete Clear AND OR NOT

Ungroup Select Image Property Query

Dokument: Übermittelt (6.38 Sek.)

VisualMOQL

Define Image Property

Simple Color Matching Color Histogram Matchir

Color Similarity: 50

Value: = Magdeburg

Attribute	Value
title	
publisher	
createYear	=1999.0
description	
country	
city	=Magdeburg

Add Delete Clear Colors

Done Cancel

Java Applet Window

Image properties

Relation definition

Define Relation

Please define the relationship between object C103 and C104

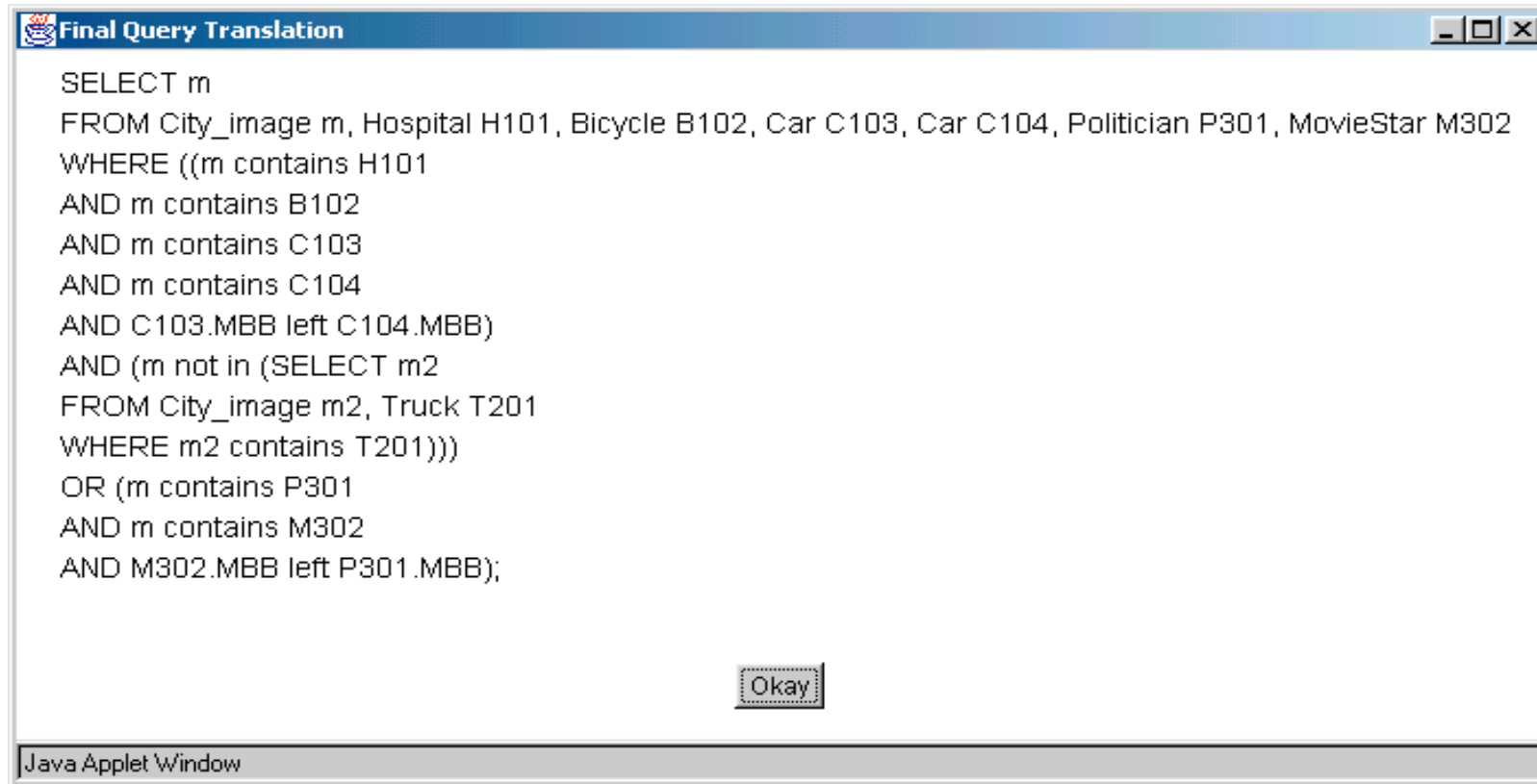
Spatial Relationship: X Y Z

Distance: X Y Z

Done Cancel

Java Applet Window

VisualMOQL



```
SELECT m
FROM City_image m, Hospital H101, Bicycle B102, Car C103, Car C104, Politician P301, MovieStar M302
WHERE ((m contains H101
AND m contains B102
AND m contains C103
AND m contains C104
AND C103.MBB left C104.MBB)
AND (m not in (SELECT m2
FROM City_image m2, Truck T201
WHERE m2 contains T201)))
OR (m contains P301
AND m contains M302
AND M302.MBB left P301.MBB);
```

Okay

Java Applet Window

Default presentation: list of files with thumbnails