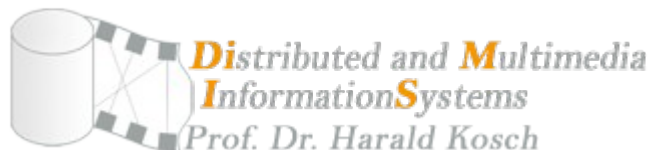


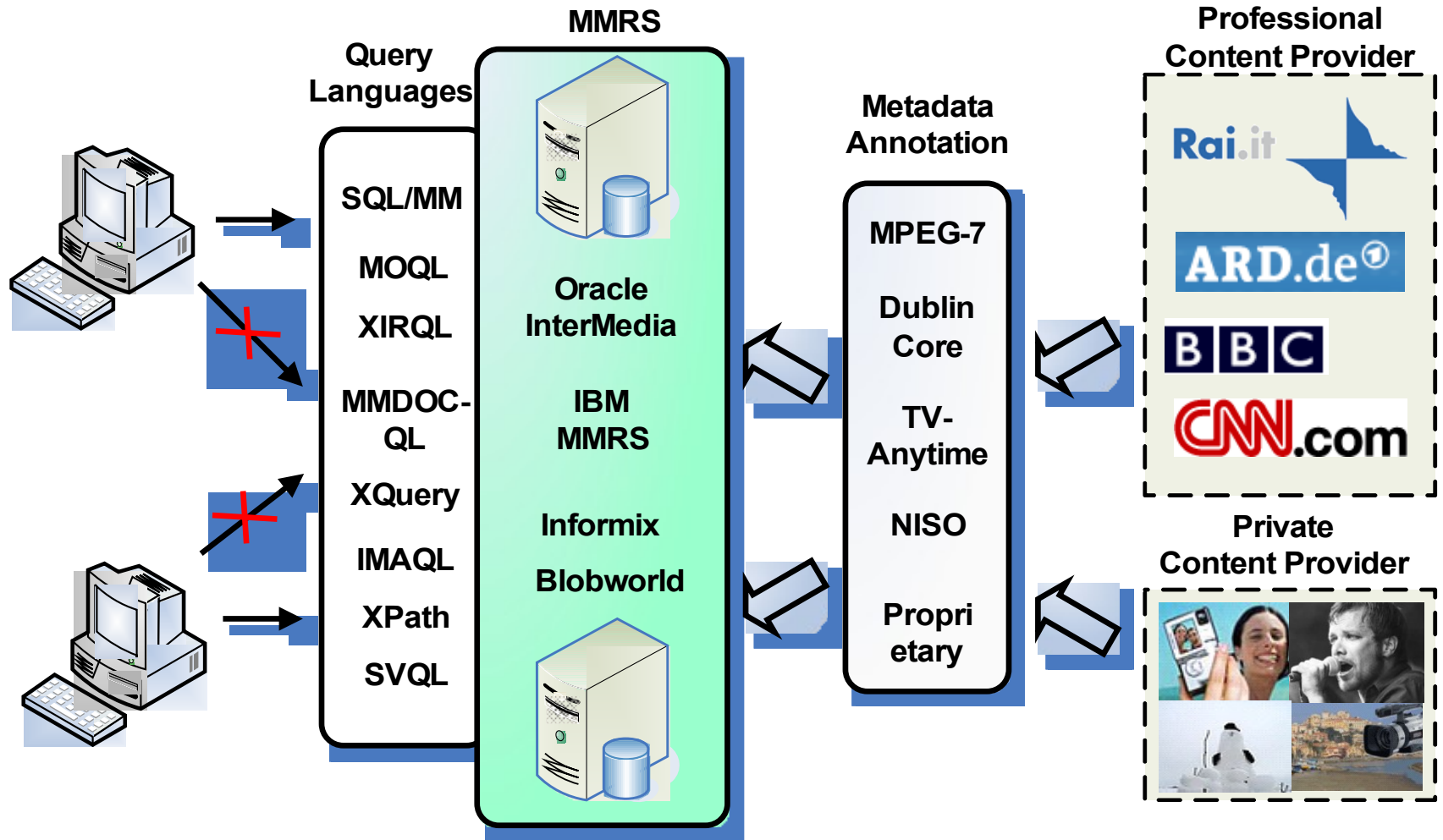
# Standardized Multimedia Retrieval in Distributed Heterogenous Database Systems

Dr. Mario Döller

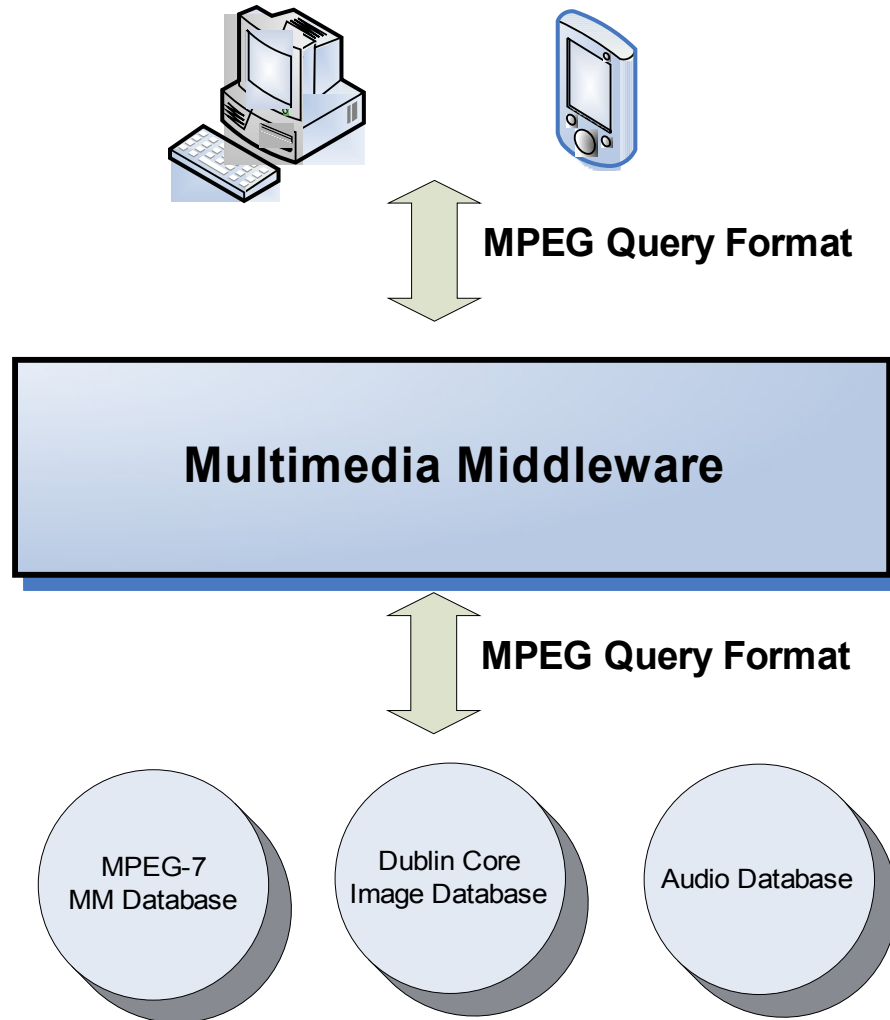


# Motivation

## Current Situation



# Standardized Retrieval in Distributed MM Systems



---

# *The MPEG Query Format*



# *The MPEG Query Format (MPQF)*

---

- Becomes International Standard Spring 2009
- General Concepts
  - bases on XML and is defined by an XML Schema
  - decoupled from any other metadata standard (also MPEG-7)
  - support for any XML based MM metadata description
  - integration of limited XQuery functionality
  - MPQF divided into 3 main categories
    - Management
    - Input Query Format
    - Output Query Format

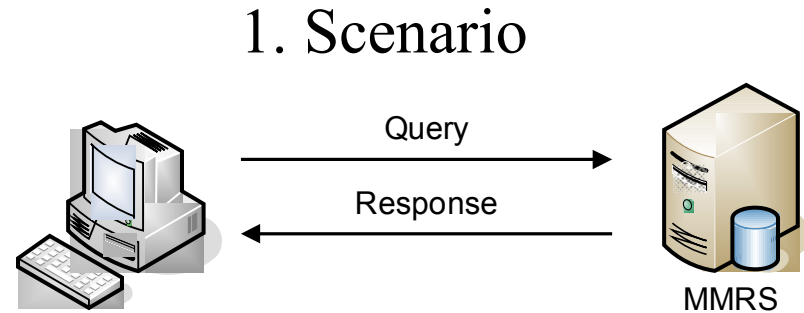


# MPQF Concepts Management

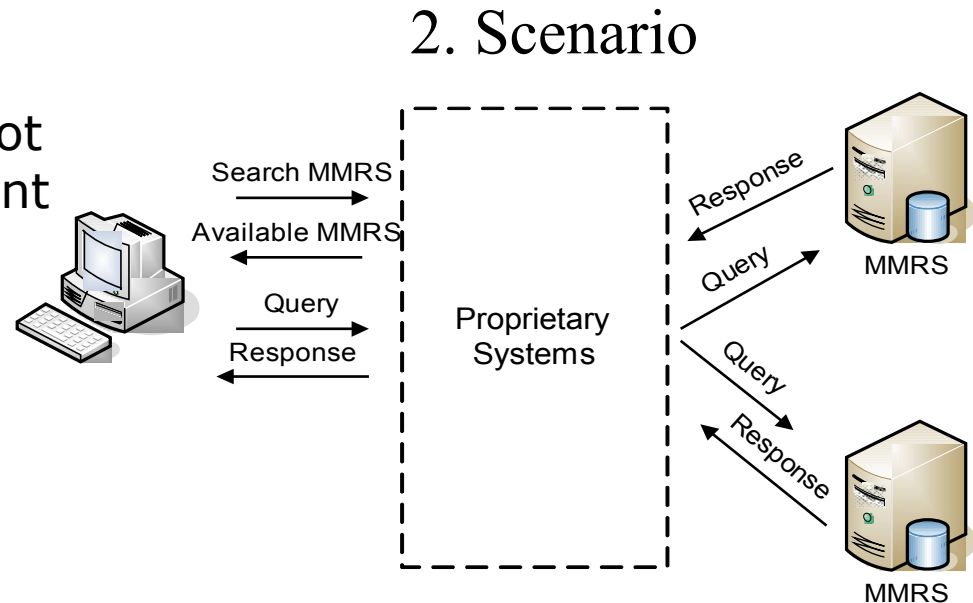
- How to find MPQF based MM Retrieval Systems (MMRSs)?

- 2 Scenarios

- MMRS is known to the Client



- MMRS(s) is/are not known to the Client

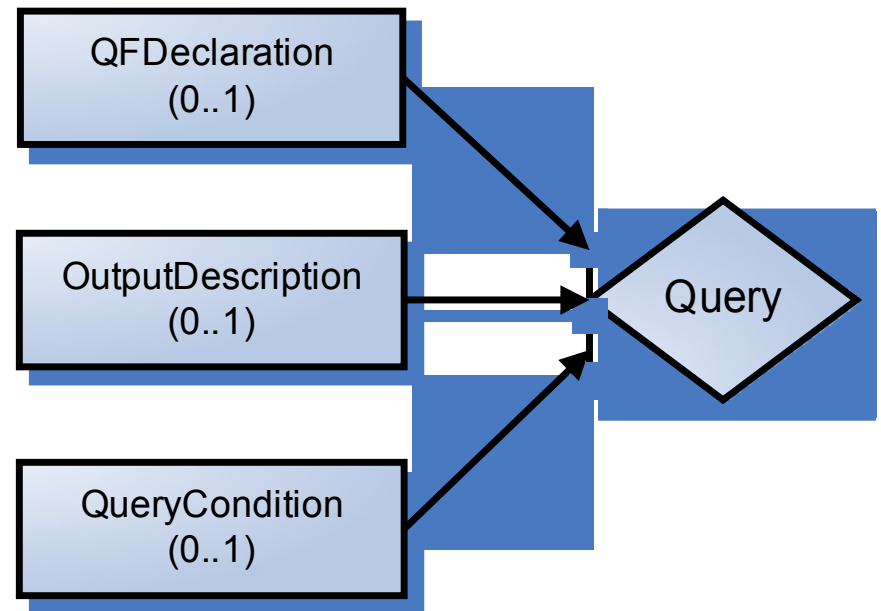


# MPQF Concepts

## Query I

- How to query MMRS satisfactorily?
- MPQF supports:
  - Synchronous/Asynchronous mode
  - Timeout functionality

### Query Design



# MPQF Concepts

## Query II

---

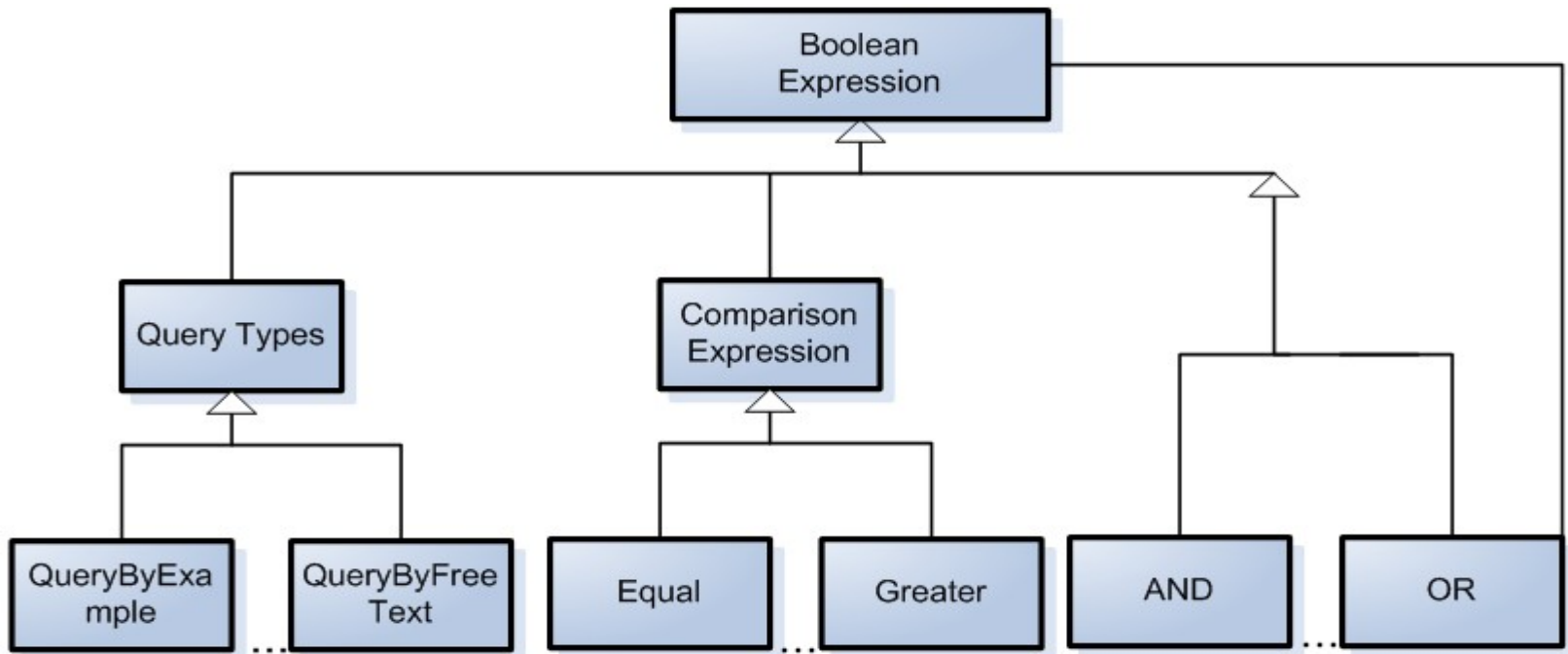
- QFDeclaration
  - declaration of resources for query conditions
    - resource can be: (structured) text, raw media or any metadata description type (e.g., DominantColorType of MPEG-7)
- OutputDescription
  - defines the content as well as structure of the result set
  - uses XPath for selecting desired types/elements of the supported metadata description
    - supports absolute and relative addressing
  - Metadata description independent
  - supports grouping/sorting
  - limitation/paging of result set





# MPQF Concepts

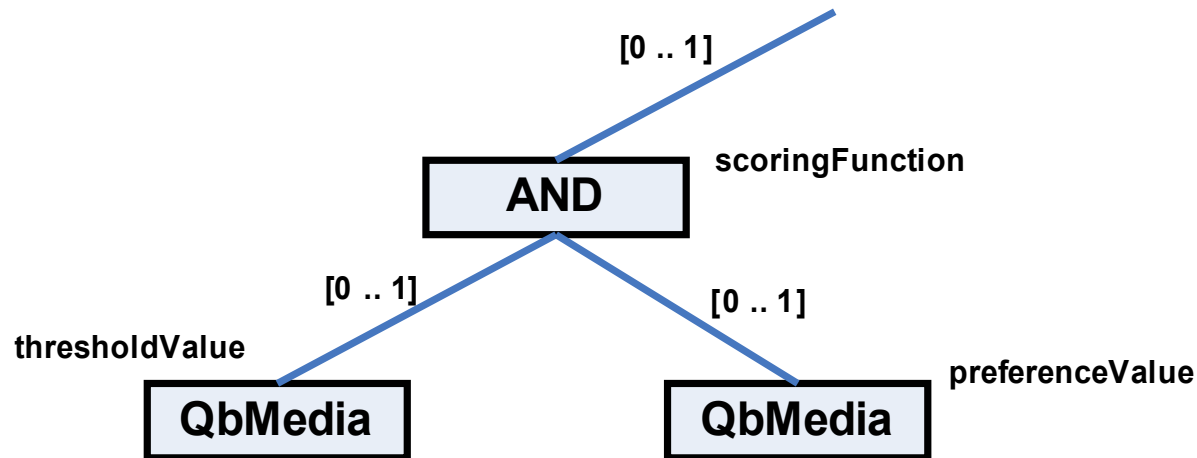
## Query - Condition



- modular filter architecture
- filter data according to *TargetMediaType*
- join functionality
- specification of granularity

# MPQF Concepts

## Query - Condition



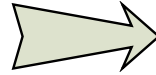
- assign *preferenceValue* and *thresholdValue* to every condition
- assign *scoringFunction* to every „Boolean Operator“ (AND, OR, XOR) (recommended to follow t-norm, t-conorm rules)
- result in rank and confidence evaluation for every item



# MPQF Examples

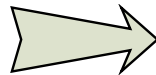
## Management I

- request: give me all available MMRS!



```
<MpegQuery>
  <Management>
    </Input>
  </Management>
</MpegQuery>
```

- request: give me all available MMRS fitting to my desired requirements



```
<MpegQuery>
  <Management>
    <Input>
      <DesiredCapability>
        <SupportedMetadata href="urn:mpeg:mpeg7:schema:2004" />
        <SupportedQueryTypes href="QueryByMedia" />
        <SupportedQueryTypes href="QueryByFreeText" />
      </DesiredCapability>
    </Input>
  </Management>
</MpegQuery>
```

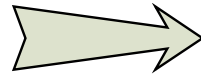


# MPQF Examples

## Query I

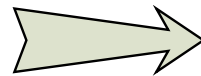
---

- Browsing Query



```
<MpegQuery>
  <Query>
    <Input />
  </Query>
</MpegQuery>
```

- QueryByFree Text



```
<MpegQuery>
  <Query>
    <Input>
      <QueryCondition>
        <Condition xsi:type="QueryByFreeText">
          <FreeText>This is a free text query</FreeText>
        </Condition>
      </QueryCondition>
    </Input>
  </Query>
</MpegQuery>
```



---

# *Multimedia Middleware*



# Existing Frameworks

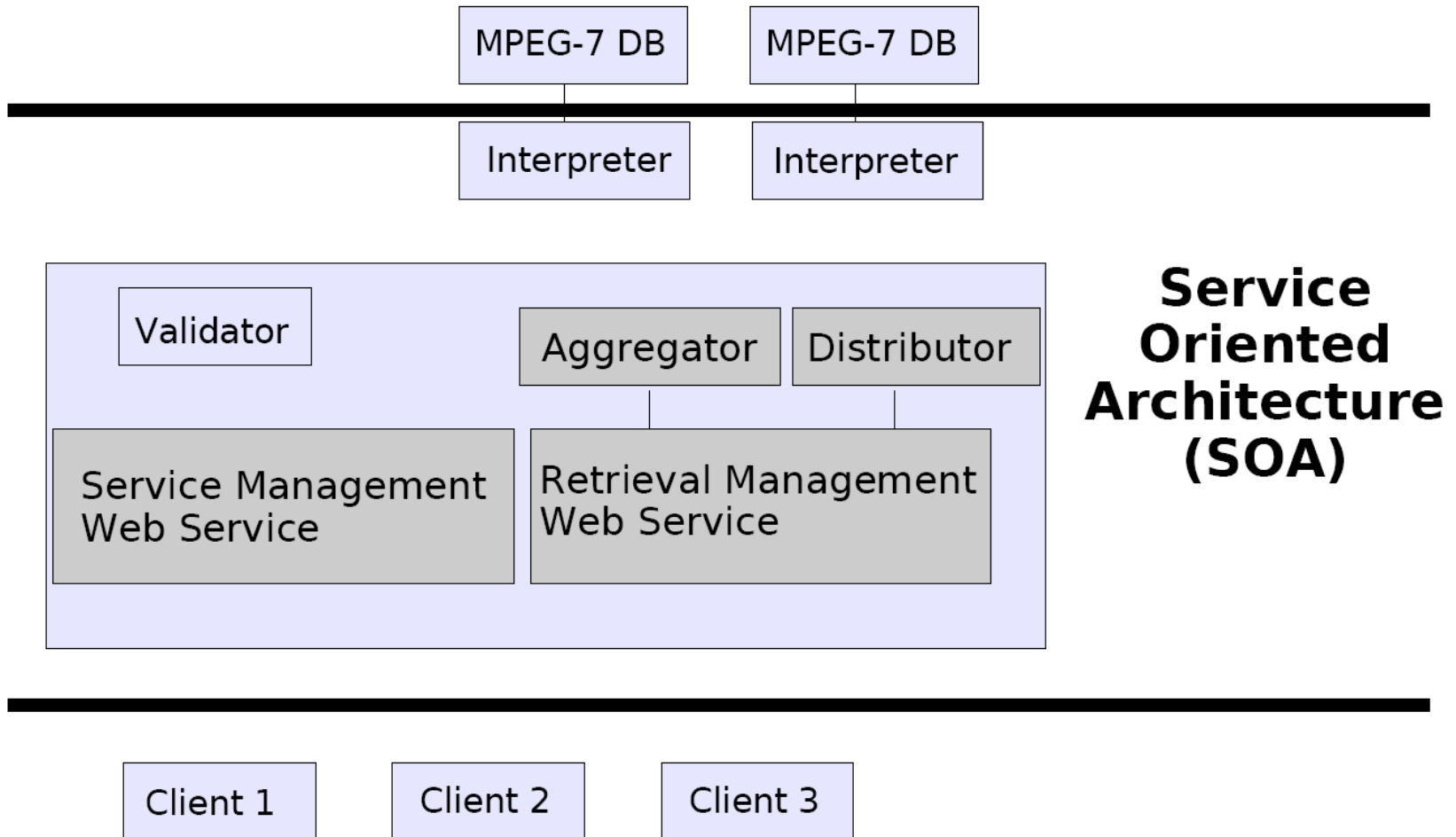
---

- Specifications
  - RMI (Remote Method Invocation)
    - Communication protocol, Java
  - CORBA (Common Object Request Broker Architecture)
    - platform-comprehensive protocols and services
- Frameworks:
  - dLIMIT
  - Framework of the HERON project
  - MOCHA (Middleware based On a Code shipping Architecture)



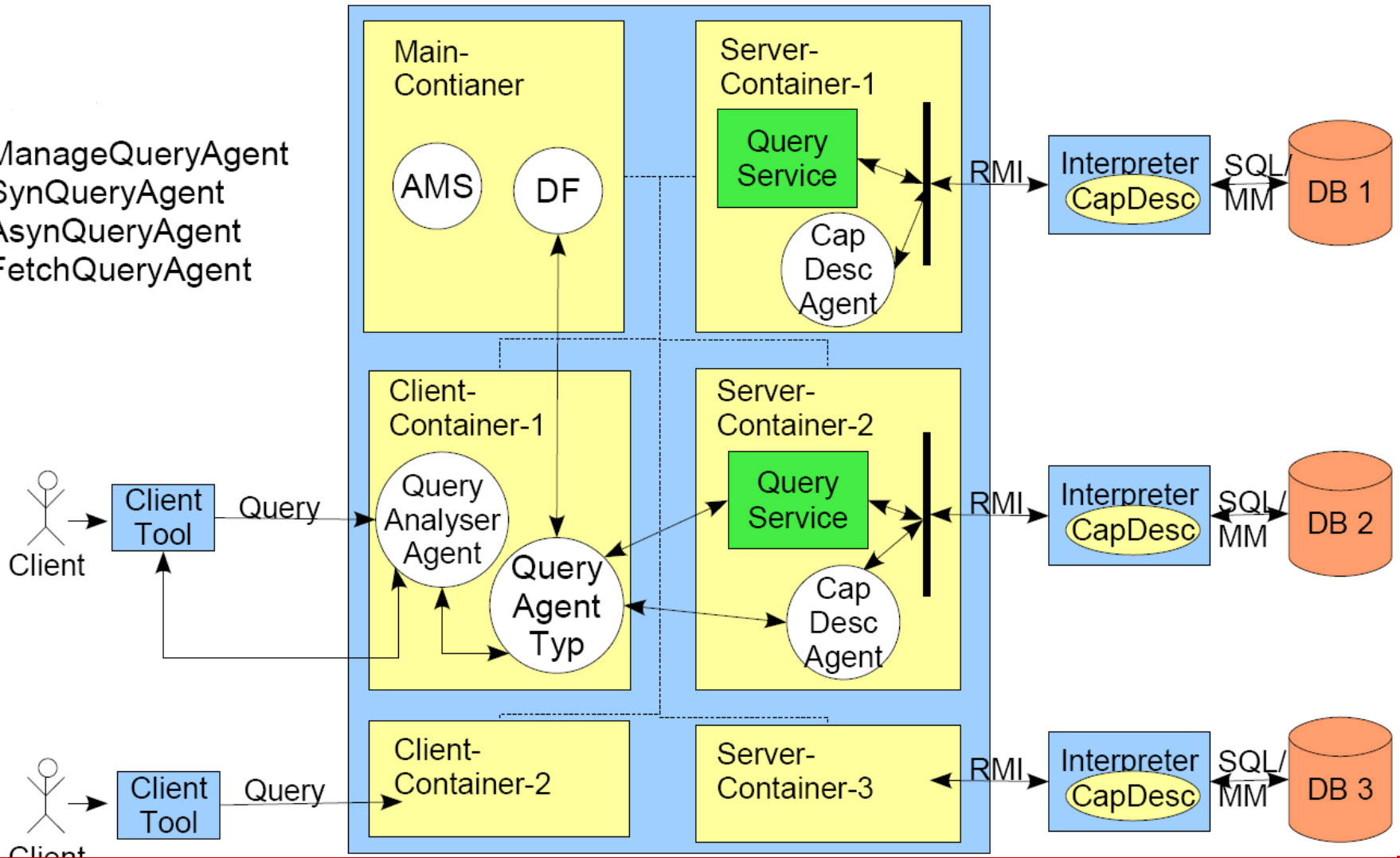
# MPQF Web Service based MM Middleware

---



# MPQF Mobile Agent based MM Middleware

- ManageQueryAgent
- SynQueryAgent
- AsynQueryAgent
- FetchQueryAgent





# *Result Aggregation – Existing Algorithm*

---

## **Identic Content**

- Combination techniques: Min, Max, CombSum, CombMNZ
- Weighted combination techniques
- Borda-fuse voting model

## **Disjoint Content**

- Raw score merging
- Round Robin
- Reference statistics
- Feature distance ranking algorithms
- Cross rank similarity comparison
- Query-based sampling

## **Overlapping Content**

- Data fusion techniques
- Shadow Document Method



# *Applicable Result Aggregation Algorithm*

---

## **Multimedia-Databases:**

- Degree of overlap not known, different content (image, audio, etc.)
- uncooperativ
- Result: list containing rank, record number, confidence information, ...

## **Applicable Algorithms:**

- (weighted) combination techniques
- Borda-fuse voting model
- Shadow document method
- Round Robin:
  - Precondition: no elimination of duplicates
  - Problem: leads to duplicated elements in the result set



---

# *Multimedia Database*



# MM Metadata Description

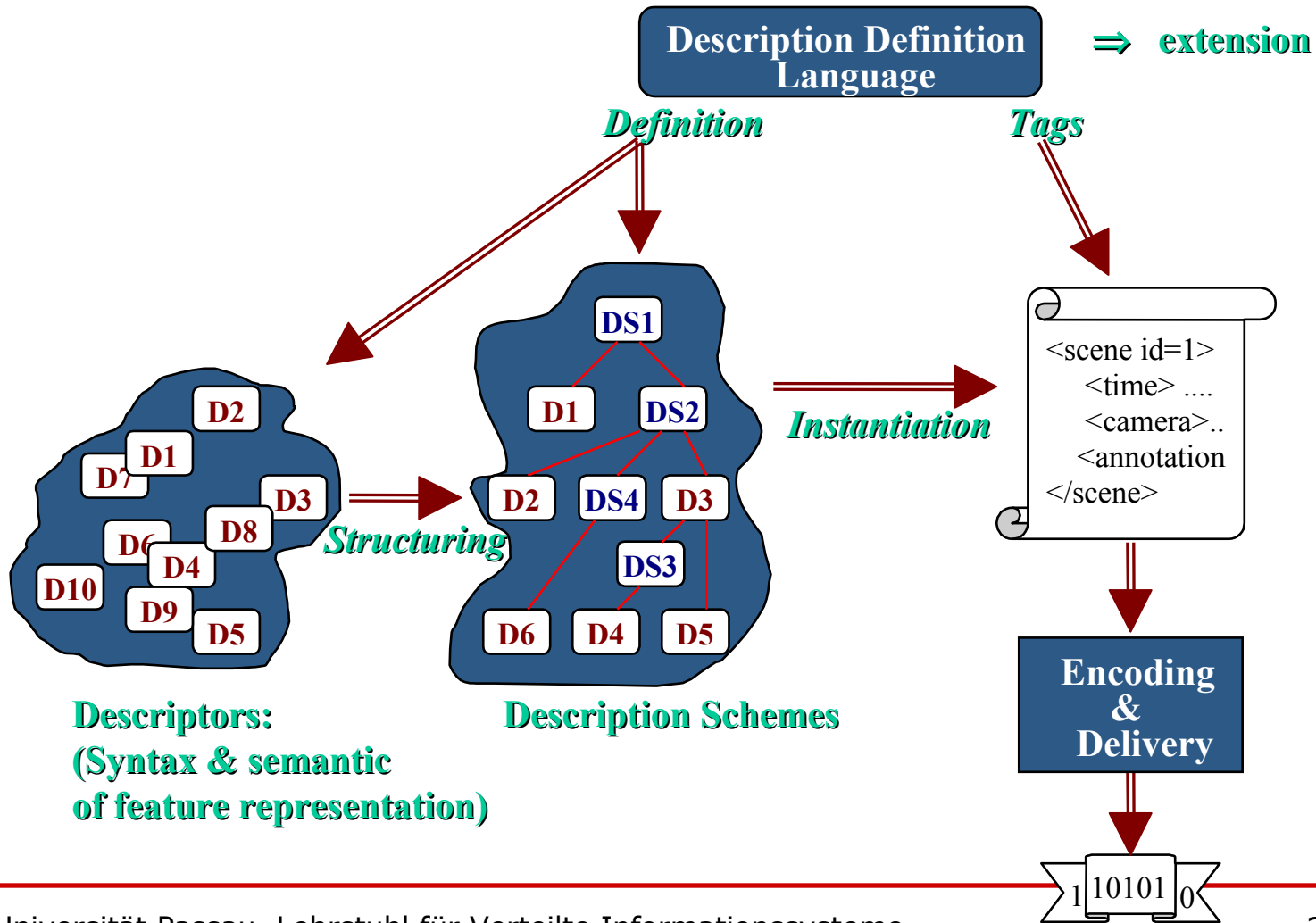
## MPEG-7

---

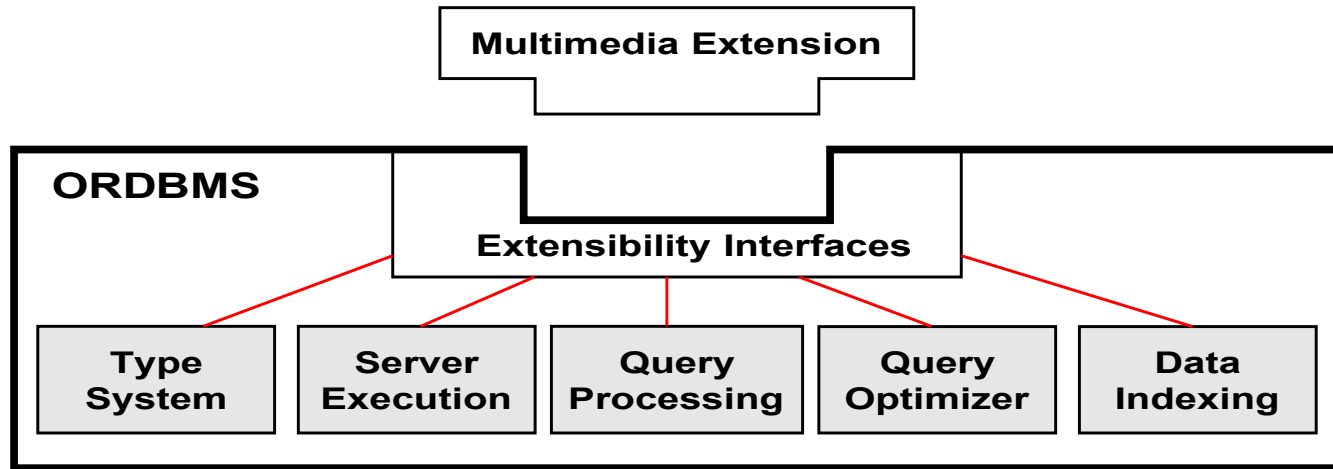
- concentrates on describing multimedia content in a semantically rich manner.
- ISO / IEC 15938 - 1: Systems
- ISO / IEC 15938 - 2: Description Definition Language
- ISO / IEC 15938 - 3: Visual
- ISO / IEC 15938 - 4: Audio
- ISO / IEC 15938 - 5: Multimedia Description Schemes (MDS)
- ISO / IEC 15938 - 6: Reference Software
- ISO / IEC 15938 - 7: Conformance
- ISO / IEC 15938 - 8: Extraction and Use
- ISO / IEC 15938 - 9: *Profile*
- ISO / IEC 15938 - 10: *Schema Definition*
- ISO / IEC 15938 - 11: MPEG-7 Schema Profiles
- ISO / IEC 15938 - 12: MPEG Query Format



# MPEG 7 Network



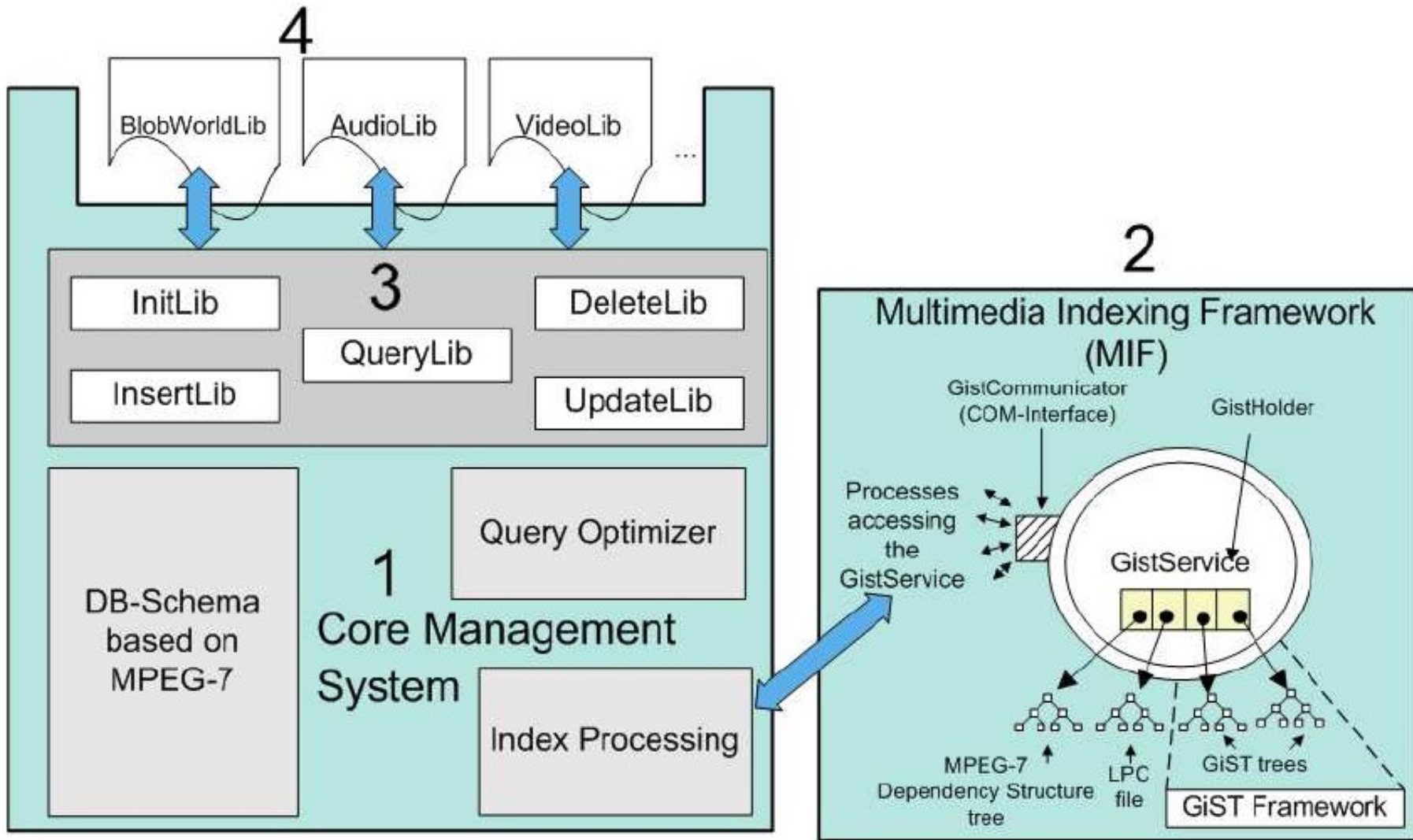
# Introduction to Extensible ORDBMS



- currently available systems
  - **Oracle Data Cartridges**
  - Informix DataBlades
  - DB2 Extenders



# MPEG-7 MMDB Architecture



# Multimedia Schema based on MPEG-7

---

- approach
  - selected MPEG-7 types become DB types / tables
  - use of XMLType to reduce the amount of datatypes in the database
  - reduce inheritance hierarchy through skipping abstract types manually
  - navigation via references (top-down) and keys (bottom-up)





## *Multimedia Schema based on MPEG-7(2)*

---

- relational keys (DOC\_ID, PART\_ID)
  - to distinguish data from various MPEG-7 files
  - to store several occurrence of the same element in the MPEG-7 document
- xsi:type references
- collections:
  - table of type
  - table of REF type
- MPEG-7 attributes/elements are mapped to DB table attributes
- Metadata for insertion process



# Multimedia Schema based on MPEG-7 (3)

```

<!-- Definition of StillRegion DS -->
<complexType name="StillRegionType">
  <complexContent>
    <extension base="mpeg7:SegmentType">
      <sequence>
        ...
        <choice minOccurs="0">
          <element name="SpatialLocator"
            type="mpeg7:RegionLocatorType"/>
          <element name="SpatialMask"
            type="mpeg7:SpatialMaskType"/>
        </choice>
        ...
        <choice minOccurs="0" maxOccurs="unbounded">
          <element name="VisualDescriptor"
            type="mpeg7:VisualDType"/>
          <element name="VisualDescriptionScheme"
            type="mpeg7:VisualDSType"/>
          <element name="GridLayoutDescriptors"
            type="mpeg7:GridLayoutType"/>
        </choice>
        ...
      </sequence>
    </extension>
  </complexContent>
</complexType>

```

StillRegionType
DOC_ID : Integer
PART_ID : Integer
MediaLocator : SYS.XMLType
TextAnnotation : dummyNTType83
SpatialLocator : SYS.XMLType
SpatialMask : SYS.XMLType
dummyAttr23 : dummyNTType27
Image()
StillRegion()

dummyNTType83

dummyNTType27

TextAnnotationType

DOC_ID : Integer
PART_ID : Integer
dummyAttr12 : dummyNTType13
relevance : Float(126)
confidence : Float(126)
lang : Varchar2(50)
TextAnnotation()

dummyType28

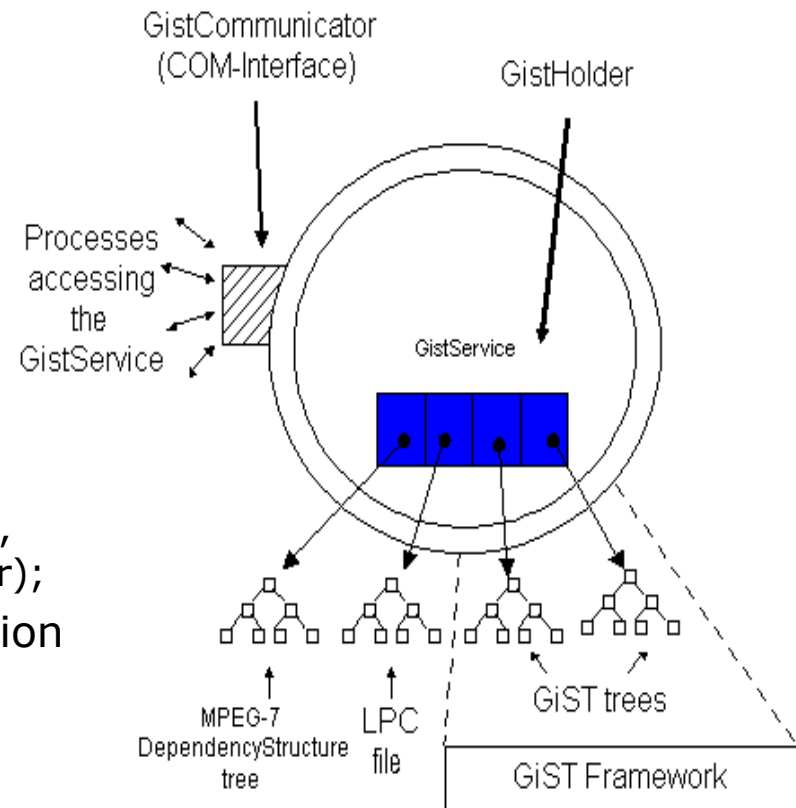
VisualDescriptor : Varchar2(60)
VisualDescriptor_PART_ID : Integer
VisualDescriptionScheme : SYS.XMLType
GridLayoutDescriptors : SYS.XMLType



# Multimedia Indexing Framework

Main features:

- GistService
  - GiST
- GistWrapper
- Multimedia Index Type
  - consists of several
    - index types,
    - operators
      - `rt_nearest_point(CLOB, CLOB, number, number);`
    - appropriate implementation (objects)



# *Multimedia Query Optimizer*

---

- enhance cost based query optimizer
  - cost (k-nearest neighbor searches)
    - implementation relies on a model for kNN query cost by Ju-Hong Lee, Guang-Ho Cha and Chin-Wan Chung
  - selectivity (range searches)



# Query Optimizer - Selectivity

---

- approach:
  - cluster the data set (feature vectors) with a density based cluster algorithm (CURE, DBSCAN)
  - calculate the density factor for every cluster ->  $\#points/V(MBR)$
  - compute selectivity for a point by identifying the correct cluster (intersect operation with MBR) and multiply the density factor with the volume of the range search



# Querying MPEG-7 MMDB

---

- query traditionally with the help of XPATH expressions:

```
SELECT extract(medialocator, '/MediaLocator/MediaUri/text()')  
FROM audio;
```

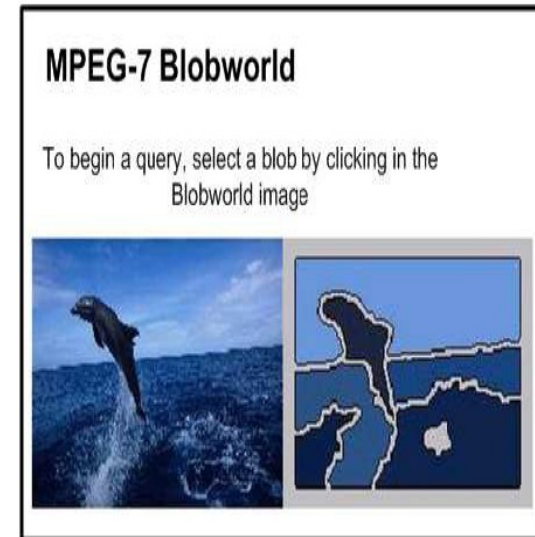
- query the database with Oracles XMLDB functionality to produce well formed MPEG-7 output
- query the database with the help of our internal QueryLib



# Content Based Image Retrieval

## MPEG-7 - Blobworld

- original system was introduced by Prof. Hellerstein and his group at the University of California, Berkeley
- **improvements:**
  - images and their blobs are described with MPEG-7 and stored and indexed by our MPEG-7 MMDB
  - providing means for enhancing the image pool at runtime
- retrieval bases on kNN search (MIF) of following weighted features: color, shape and texture
- result is returned as MPEG-7 document



# *Audio Recognition Tool*

---

- developed in connection with RWTH Aachen (Holger Crysandt)
- extracted MPEG-7 descriptions of audio files are stored and indexed by the MPEG-7 MMDB
- retrieval allows:
  - exact match search by interpreter, genre or song title
  - similarity search that relies on audio signature



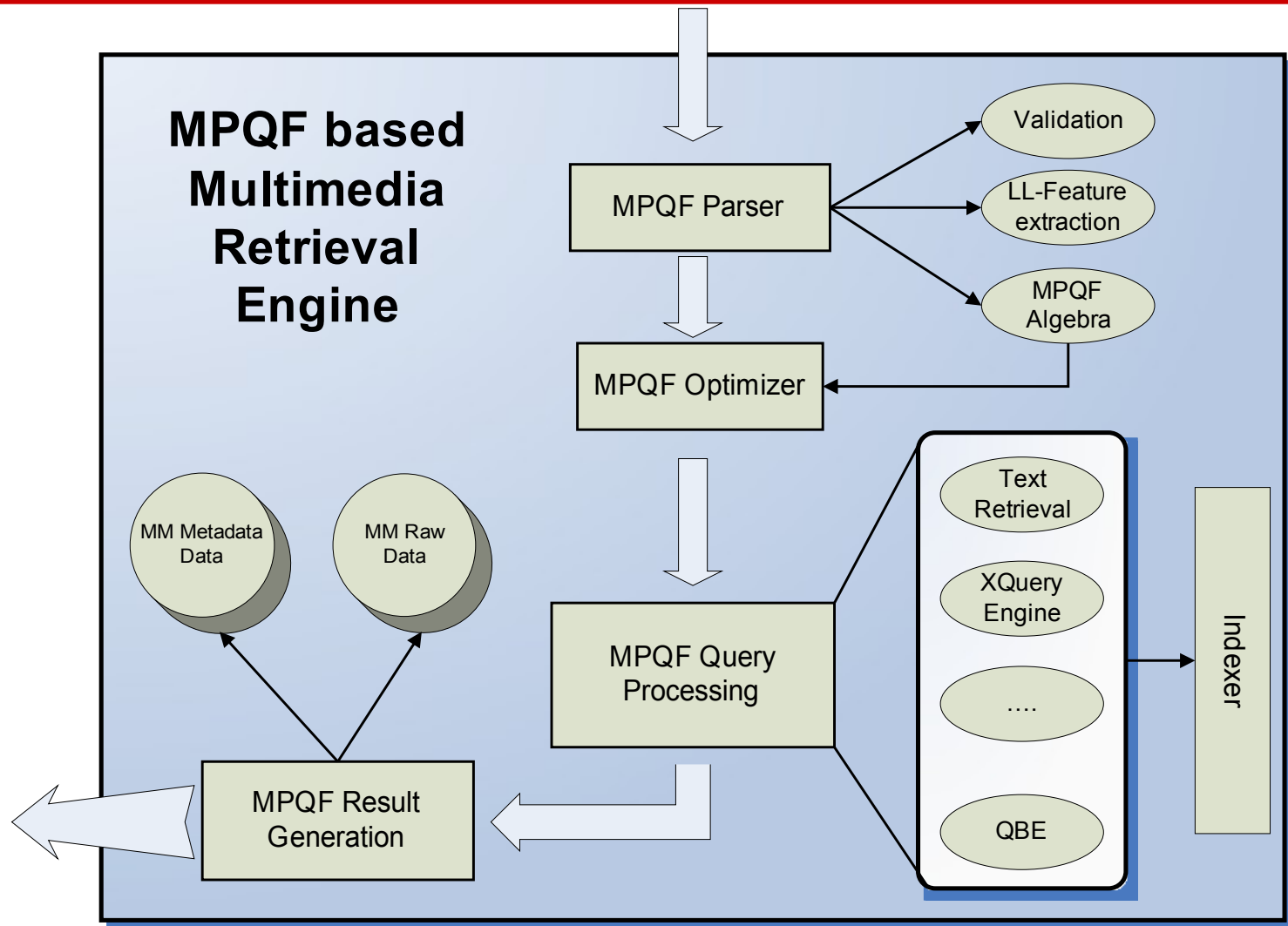


---

# *Outlook*



# MPQF MM Database System



# Outlook

---

- Future Work
  - 88. MPEG Meeting (April 2008)
    - Reference Software
  - JPEG
    - Decided to use MPQF as query language for the JPSearch project
  - MPQF is used in the following EU projects
    - SAPIR (<http://www.sapir.eu/>)
    - PHAROS (<http://www.pharos-audiovisual-search.eu/>)
    - ENTHRONE (<http://www.ist-enthroned.org/>)
  - Research
    - Develop an MPQF Retrieval Engine
    - Develop an MPQF Service Aggregation
    - Develop applications using MPQF Retrieval Engine and/or JPSearch environment



---

# Questions?

