Chapter 5: Multimedia Database System

• Design and Architecture of a Multimedia database
• Indexing and organizing multimedia data
Multimedia Architecture

• Multimedia Architecture Requirements
  – ACID test
  – Multimedia Server Requirements
• Distributed Multimedia System
  – Super server concept
• Client-Server Systems
• P2P
• Media Streams
<table>
<thead>
<tr>
<th>id</th>
<th>size</th>
<th>fps</th>
<th>title</th>
<th>filename</th>
</tr>
</thead>
<tbody>
<tr>
<td>000001</td>
<td>530M</td>
<td>30</td>
<td>soam</td>
<td>l6.mpg</td>
</tr>
<tr>
<td>000002</td>
<td>450M</td>
<td>30</td>
<td>tibor</td>
<td>l7.mpg</td>
</tr>
<tr>
<td>000003</td>
<td>600M</td>
<td>30</td>
<td>parag</td>
<td>l5.mpg</td>
</tr>
<tr>
<td>000004</td>
<td>510M</td>
<td>30</td>
<td>wei</td>
<td>l4.mpg</td>
</tr>
</tbody>
</table>
Semi-intergrated
Fully Intergrated

index

buffers

storage

chapter5: Multimedia Database System
Multimedia Architecture Requirements

• Database architecture as a structure that facilities the database to complete a transaction

• Four basic properties that a transaction should posses
  – Atomicity: All or nothing property. A transaction is an indivisible unit that is either performed or not
  – Consistency: A transaction must transform the database from one consistent state to another consistent state
  – Independence: Transactions execute independently of one another
  – Durability: The effects of a successfully committed transaction should be permanently recorded in the database

  – ACID test of transaction reliability
ACID test

• For a single-user PC database where only one person is carrying out transactions at any one time the circumstances for the ACID test may be irrelevant

• Important for large number of users which access the database at the same time

  – A transaction can than only be achieved by locking the data rows involved to stop other users changing the data

  – Replicated database there may be more than one copy of the data that needs to be updated at the same time
• Architecture of a multi-user database can become complex

• It is not clear which architecture would be the best option for a multimedia database
  – A transaction involving multimedia data will in general be expected to take longer
  – Locks will have to be maintained for longer periods
• Formal database architecture
• Separate user view from the system view
• Three-layer architecture
Three-layer architecture

- The external level provides the user's view of the database
  - It is a partial view

- The conceptual level is the community view of the database
  - Logical level as seen by the system administrator
  - In a relational database, relational conceptual level

- Internal level
  - The way the data is physically stored
  - In a relational database the internal level must not be relational
    - Records, pointers, etc..
• For multimedia objects, performance depends on the rate at which information can be transferred from storage memory for processing

• Block size affects the performance
  – Number of fetch operations
The architecture of the database system

- The architecture of the database system is influenced by the underlying computer and network system
  - **Centralized database** system run on a single computer system that does not interact with other computer systems
  - **Client-server system**, networking computers allow a division of work. Task relating to database structure are executed on server, presentation on the client computer
  - **Distributed database systems** have been developed to handle geographically and administratively distributed data spread over multiple computer systems
Figure 8.2  Multimedia databases - user, conceptual and physical storage views
Multimedia Server Requirements

• Often large scale applications
• Take into account:
  – User access behavior
  – Bandwidth
  – Storage requirements
    • (Complex multimedia formats)
Storage hierarchy

• Example, videos on demand:
• High popular videos are stored in storage media with the highest bandwidth
Characteristics

• Minimal response time
• Reliability and availability
• Ability to sustain *guaranteed number of streams*
• Real-time delivery
• Exploit user access patterns
Distributed Multimedia System

• In a relational database that is distributed a table may be divided into a number of subrelations
• Horizontally - fragments consists of columns but only some rows
• Vertically - fragments consists of all rows but only some columns
• Partitioning of the data
Distributed Multimedia System

• Replicate fragments so that duplicates are stored on several sites
• LOBs (video, music)) movements to a site, where they are likely to be requested (duplicates)
  – Even daily basis!
Scalability

• Increasing number of users
• Size of data objects
• Amount of accessible data
  – Search, access, management
• Non-uniform request distribution
Super server concept

• Distribute load among several servers
  – Problems arise when server selection is mainly based on systems defaults or on the user choice
  – This kind of static selection can cause uneven loads

• Dynamic server selection by alternatively mapping the servers in a local cluster

• Saves local load problems
Super server concept

• Requests are directed to an appropriate server according to the location and the requested data, the current load of the servers, the location of the servers and the available network bandwidth.

• User contacts a multimedia server as a normal server, and it makes the decision which is the most appropriate server.
Client-Server Systems

• A special case of distributed systems
  – Certain sites are designated as clients and others as servers

• Introduce
  – DataLinks as a specific art of SQL3
  – Development of intelligent middleware
DataLinks

• Store large unstructured data objects in a file system near a relational database

• Allows existing applications to incorporate multimedia with no changes to them

• Video and audio objects need to be streamed out to the client
  – Database servers do not have these capabilities
Intelligent Middleware

- Change information across systems developed by different vendors
  - Oracle, Ingres, DB2, MySQL
- Integration of information
- Three-tier systems were developed
  - Gateway to manage connections between the databases
- In large system there will be many servers
  - Data from local and external resources
3-tier System using mediator
Peer-to-Peer Networks

• Type of network in which each workstation has equivalent capabilities and responsibilities

• A peer-to-peer (P2P) application is different from traditional client-server model

• Applications act both as client and server

• P2P networks are simpler
  – Low performance under heavy load
P2P application

• No central server
  – Napster (original), Freenet
• Discovering other peers
• Querying peers for content
• Sharing content with other peers
Heterogeneous Distributed DBMS

- Homogenous system all the sites use the same DBMS system
- Heterogeneous system different DBMS, different data models
Content Management

• Integration of a number of technologies
• Degree of semantics
  – Artifacts (date, location), content information (sentence, key shape, color histogram), domain concepts like ontologies
• Decomposition of media into a database in terms of storage of metadata, building an indexing structure
  – should be an automatic process
chapter5: Multimedia Database System
Media Streams

• An important objective of multimedia systems design is to transfer data at a constant speed.

• Streaming is a technique for transferring data, so that it can be processed as a steady and continuous stream.

• By using streaming, the client browser can display the data before the entire file has been transmitted.
Definitions

• **Media stream:** the output of a sensor device such as a video, audio or motion sensor that produces a continuous or discrete signal

• **Live multimedia:** the scenario where the multimedia information is captured in a real-life setting

• **Continuous queries:** persistent queries that are issued once and then logically run continuously over live and unbounded streams
Media Streams

• If the streaming client receives the data more quickly than required, it needs to be saved in a buffer

• However if the data does not arrive quickly enough, the presentation of the data will be not smooth
Example of MMDBMS

- Digital Library
- News-On-Demand
- Video-On-Demand
- Music Database
- Telemedicine
- Geographic Information System
Nature of Multimedia Data

• Large amount of data
• Time sensitive
• Vague matching
Querying

• Image
• Audio
  – Music
  – Sound
  – Speech
• Video
Querying Image

• Common approach
  – allow query by sketches (color, shape, texture) or examples.
  – perform matching by Feature Vectors
    \[ F = (v_1, v_2, \ldots, v_n) \]
  – e.g. Color Histogram
Querying Image

• Existing Systems:
  – QBIC
  – VisualSEEK
  – PhotoBook
  – Virage
  – FourEyes
Indexing and organizing multimedia data

• Requirements
  – support spatio-temporal operations
  – support fuzzy matches
Indexing Images

• N-dimentional indices for feature vector
• Well studied in DB/CG community
• Two examples:
  – VP-tree
  – R-tree
VP-tree
VP(Vantage-point) -tree
R-tree
Indexing Audio

- Audio are modeled as strings
- Inexact match is needed
- Common indices for string search can be used
- Example
  - PAT-tree
PAT(Patricia )-tree

chapter5: Multimedia Database System
Indexing Video

• Treat time as third dimension
• We can use any multidimension indexing structures