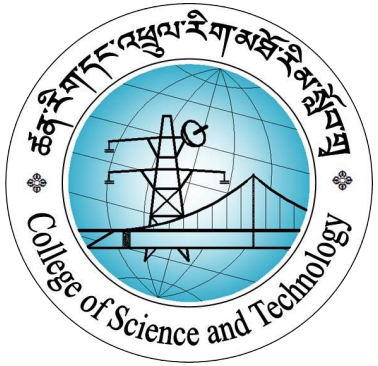

Royal University of Bhutan

DBS101 Database Systems Fundamentals

Lesson 19

Learning Outcomes

1. Describe the organization of records in files.
2. Explain the concept of a data dictionary and its storage.
3. Understand the purpose and functionality of a database buffer.

Data Dictionary Storage

- Stores "data about data" or metadata
- Contains information about relations, attributes, views, constraints, users, and more

Example: Information about the Instructor relation

Data Dictionary Storage

- Stores "data about data" or metadata
- Contains information about relations, attributes, views, constraints, users, and more

Example: Information about the Instructor relation

Information Stored in Data Dictionary

- **Names of relations and attributes**

  Example: Instructor, ID, name, dept_name, salary
- **Attribute domains and lengths**

  Example: ID is a numeric domain of length 9
- **View definitions**

  Example: Definition of view Student(ID, name) from Instructor relation

Information Stored in Data Dictionary

Integrity constraints (e.g., keys)

Example: ID is the primary key for Instructor

User information (names, passwords, authorizations)

Statistical data about relations

Example: Number of tuples in Instructor relation

Storage of Data Dictionary

Can be stored using **special data structures** and **code.**

It is preferred to be **stored as relations** in the database itself.

Example: Relation_metadata, Index_metadata relations

Data dictionary allows use of database capabilities for fast access.

Storage of Data Dictionary

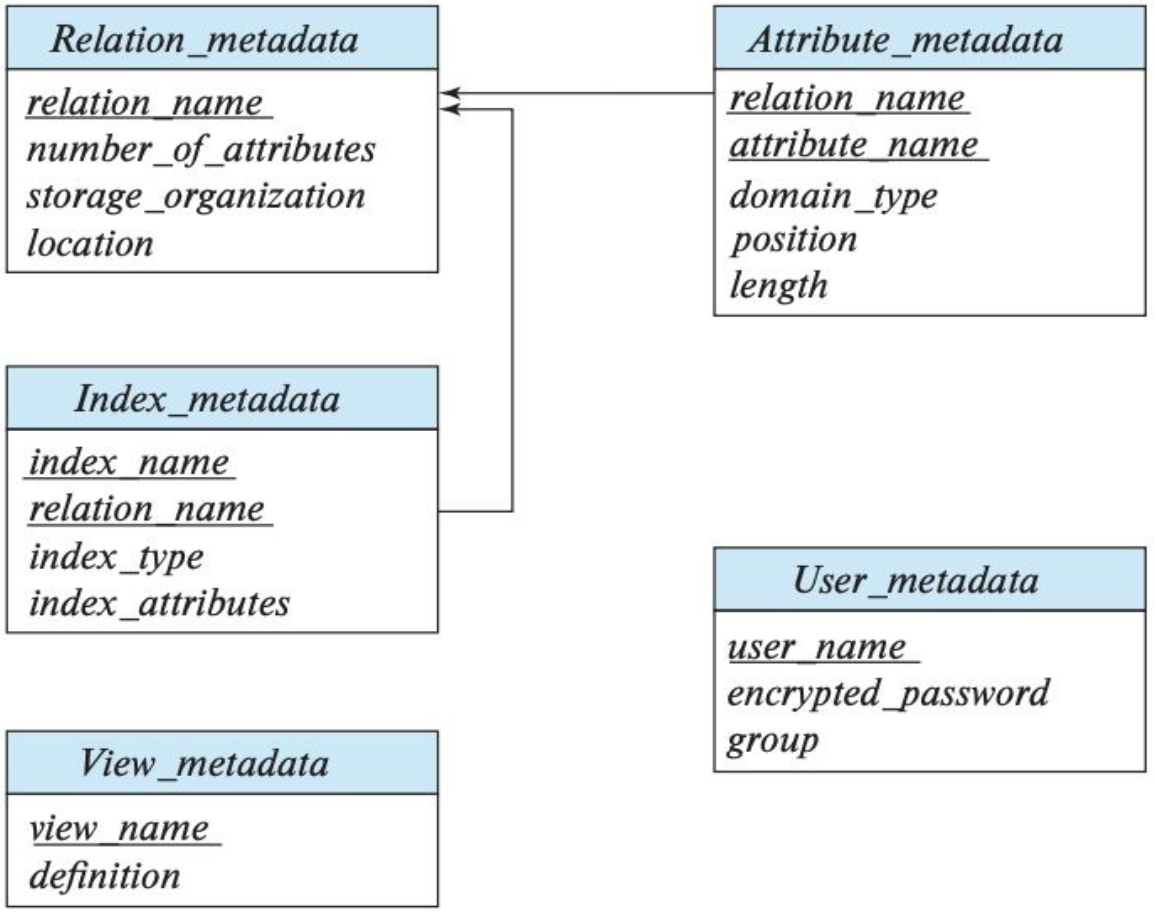

Figure 13.12 Relational schema representing part of the system metadata.

Database Buffer

Part of main memory available for **storing disk blocks.**

Buffer manager handles **allocation of buffer space.**

Example: If 1GB buffer, can hold 256K disk blocks of 4KB each

Buffer Manager

In databases, programs request blocks from buffer manager.

Example: Request for block containing Instructor(12345, ...)

If block in buffer, returns memory address

If not in buffer, reads block from disk to buffer

Buffer Manager Functions

1. **Buffer replacement strategy (e.g., LRU)**

Example: Replace **least recently used(LRU)** Instructor block

2. **Pinning/unpinning blocks**

Example: Pin Instructor block before reading tuples

3. **Shared and exclusive locks on blocks**

Example: Get shared lock on Instructor block for read

Output of modified blocks to disk

4. **Forced output of blocks**

Example: Force output of Instructor block before commit

Buffer Replacement Strategies

1. **LRU (Least Recently Used)** - replace least recently used block

Example: Replace least used Instructor block

- **Database systems can predict future block access**

- **Use knowledge of operation being performed**

Buffer Replacement Strategies

Examples of Replacement Strategies

- **Toss-immediate for temporary blocks**

  Example: Toss Instructor blocks after join with Department
- **MRU (Most Recently Used)** for blocks needed again soon
  Example: Keep most recent Department block for next Instructor tuple
- **Retain frequently accessed data** like indices, data dictionary

Factors Affecting Replacement

Concurrency control - retain blocks for active requests
Example: Keep Instructor block pinned for active transaction

Crash recovery - constraints on writing modified blocks
Example: Don't write Instructor block until all changes logged

Reordering of Writes and Recovery

- File systems may reorder writes for optimization.
  Example: Write file data before updating metadata
- Can lead to inconsistencies if crashed during reordering
- Modern file systems use journaling and log disks
  Example: First log all writes, then apply to disk
- Databases implement their own logging

Next Lesson:

Unit VII : Database Storage Structures

7.7 Column-Oriented Storage

7.8 Storage Organization in Main-Memory Databases