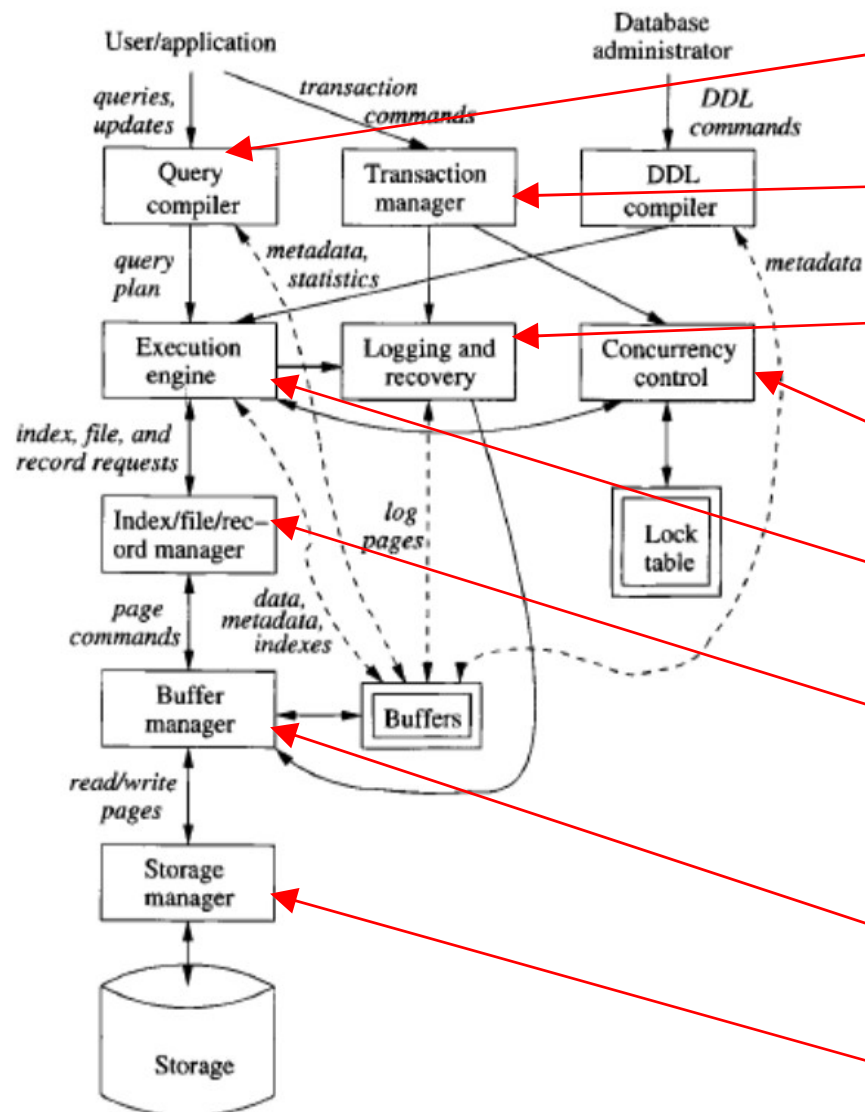


Database system implementation

5. Arranging Data on Disk

Database management system components



Query compiler translates the query into an internal form called a *query plan*

Transaction manager accepts transaction commands from an application, which tell the transaction manager when transactions begin and end

Logging and recovery manager is responsible for the durability of transactions

Concurrency-control manager is responsible for assuring atomicity and isolation of transactions

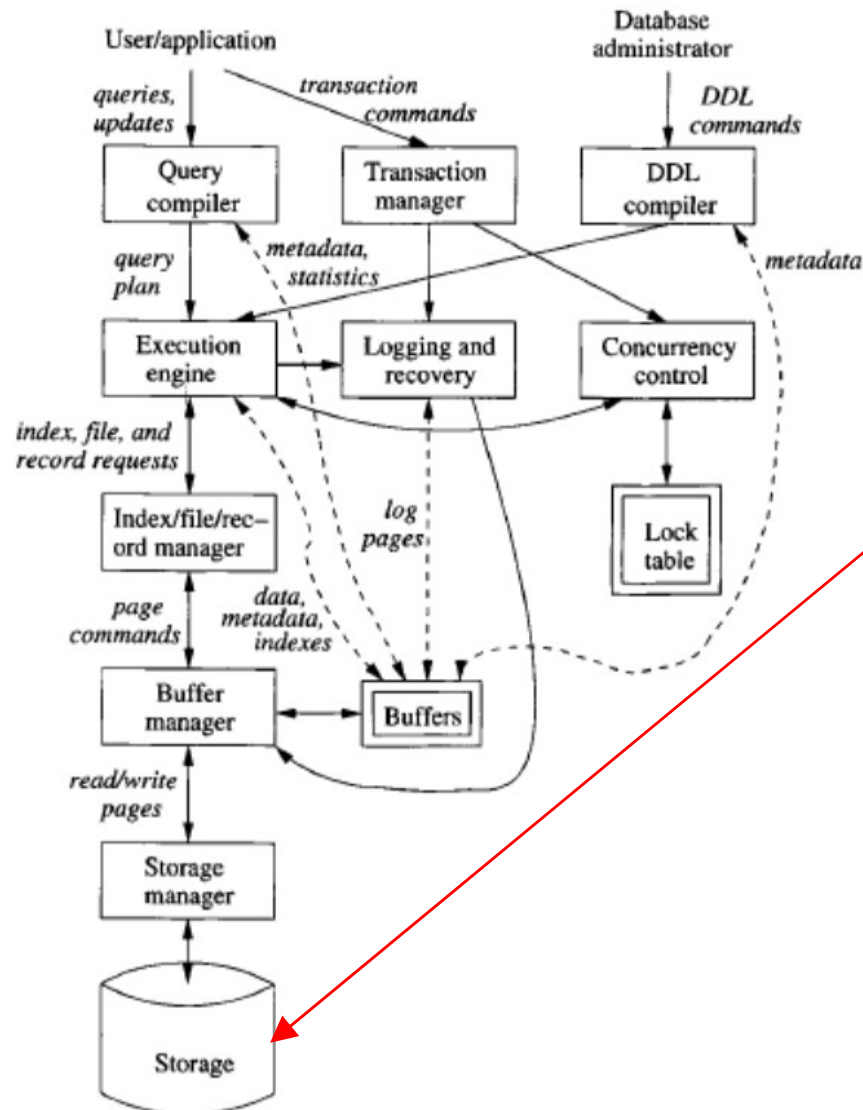
Execution engine, which has the responsibility for executing each of the steps in the chosen query plan

Index/file/record manager knows about holding relations, the format and size of records in those files, and index files, which help find elements of data files quickly

Buffer manager is responsible for partitioning the available main memory into *buffers*, which are page-sized regions into which disk blocks can be transferred

Storage manager controls the placement of data on disk and its movement between disk and main memory

Data Structures



1. *Data*: the contents of the database itself
2. *Metadata*: the database schema that describes the structure of the database
3. *Log Records*: information about recent changes to the database; these support durability of the database
4. *Statistics*: information gathered and stored by the DBMS about data properties such as the sizes of, and values in, various relations
5. *Indexes*: data structures that support efficient access to the data

Database schema (metadata)

- Database schema is a collection of relation schemas.
- Database schema is stored on the disk in the *Data dictionary*

Schema of S (Suppliers) relation

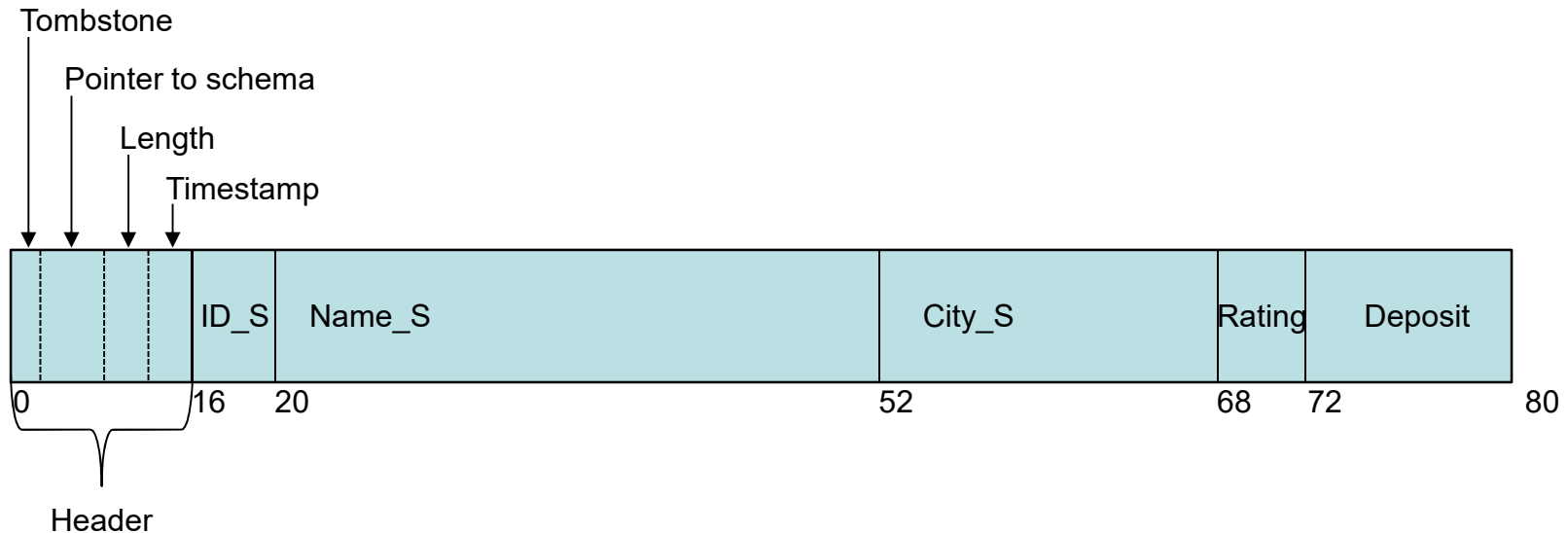
Attributes of S (Suppliers)

Identifier	Type	Length (bytes)	Restriction	Order number
City_S	Char	16		3
Deposit	Float	8		5
ID_S	Integer	4	Primary key	1
Name_S	Char	32		2
Rating	Integer	4		4

Fields, Records and Blocks

- *Field* is a sequence of bytes representing an attribute
- *Record* is a sequence of fields representing tuple
- *Block (disk page)* is a sequence of bytes with fixed length which includes several records of one relation
- Relation is represented by a chained list of several blocks

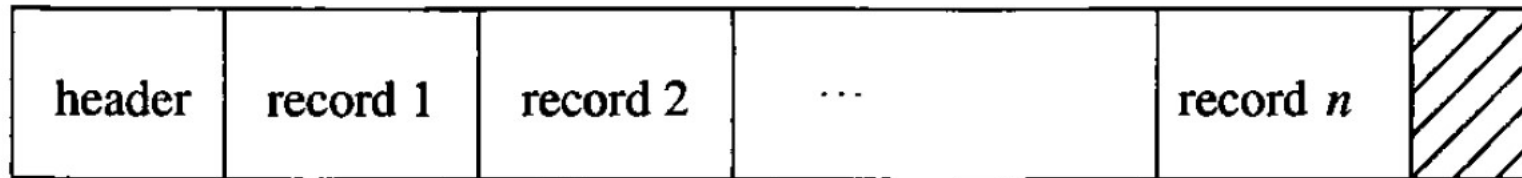
Layout of record for tuple



Record header keeps the following additional information:

- *Tombstone*: 0 – record is not deleted; 1 – record is deleted
- *Pointer to schema*: pointer to a place where the DBMS stores the schema for relation S
- *Length*: the length of the record
- *Timestamp*: the time the record was last modified

Block holding records



- Records representing tuples of a relation are stored in *blocks* (pages) of the disk
- All blocks have fixed length (4-64 KB in different DBMS)
- If we need to access or update the record, we should move entire block containing this record into buffers in main memory
- One relation can occupies several block

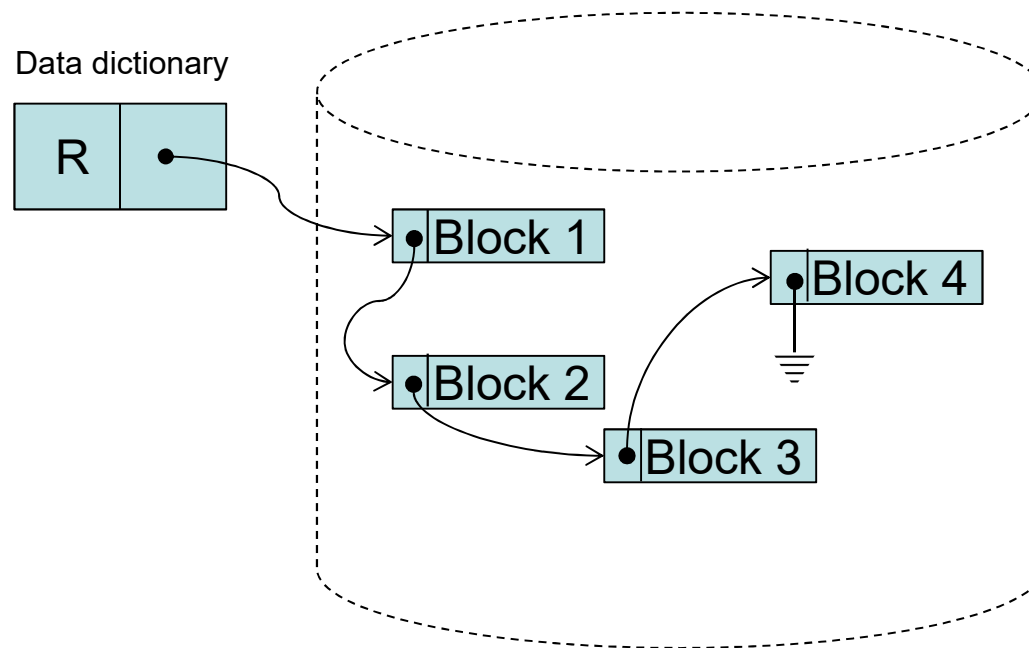


Block header

- Link to the next block (if the relation uses several blocks)
- Information about which relation the tuples of this block belong to
- Timestamp indicating the time of the block's last modification



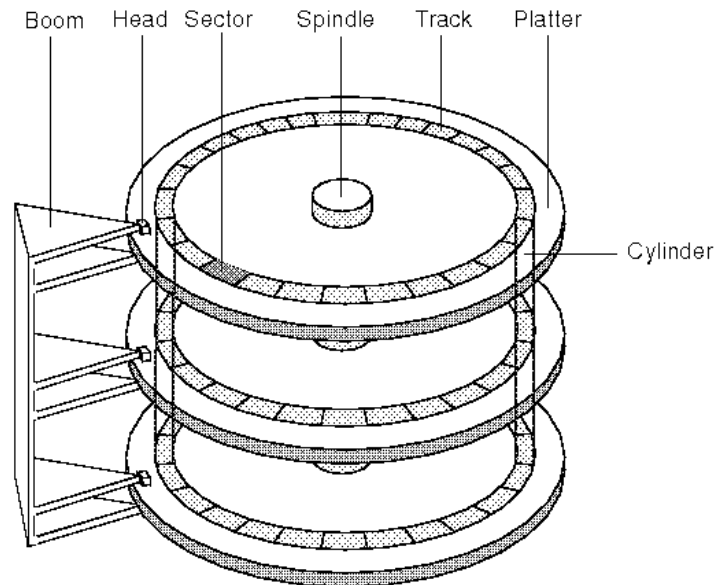
Chained list of blocks representing relation R



Record address

1. The number of the cylinder of the disk
2. The number of the track within the cylinder
3. The number of the sector (block) within the track
4. The offset of the beginning of the record within the sector

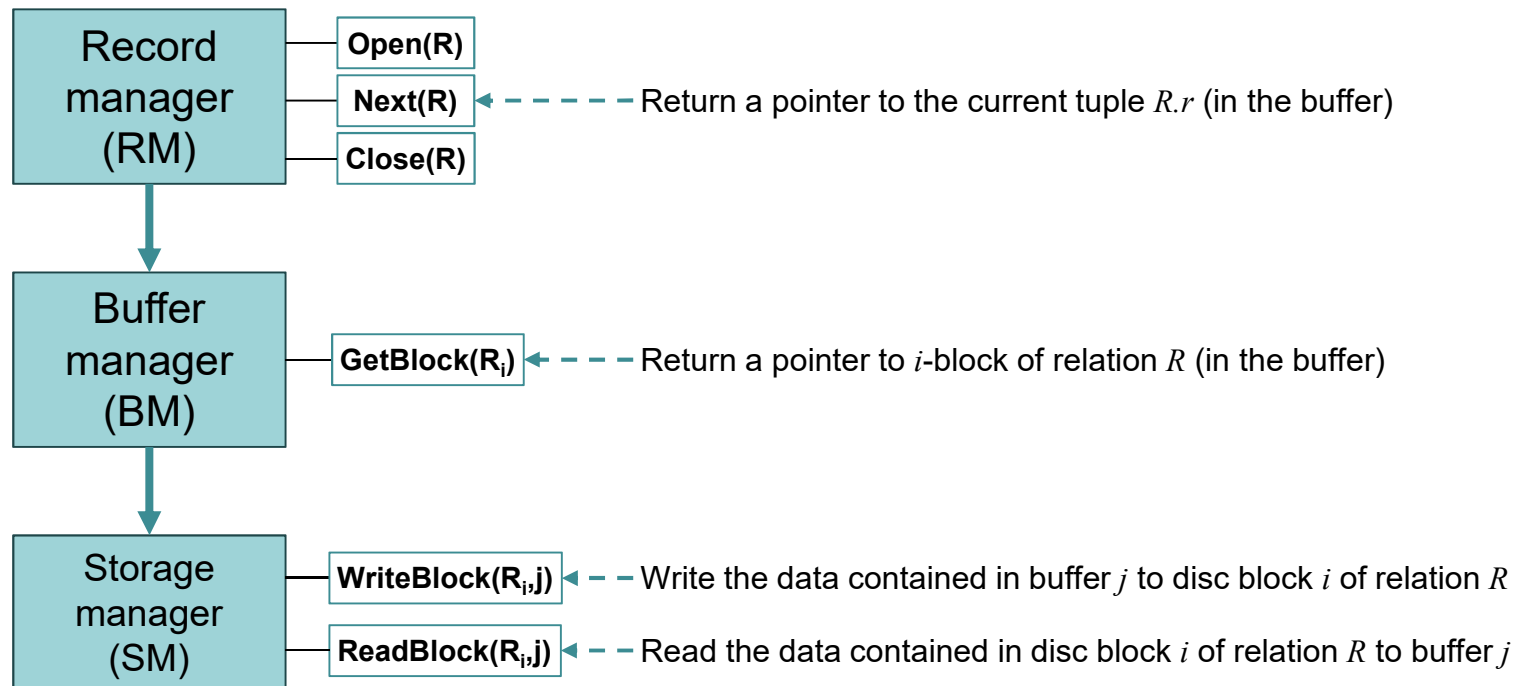
Memory hierarchy and Disks



Homework: Database Systems: The Complete Book

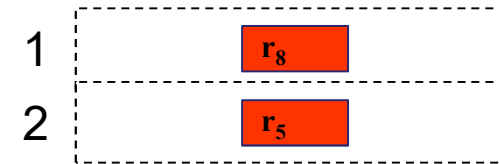
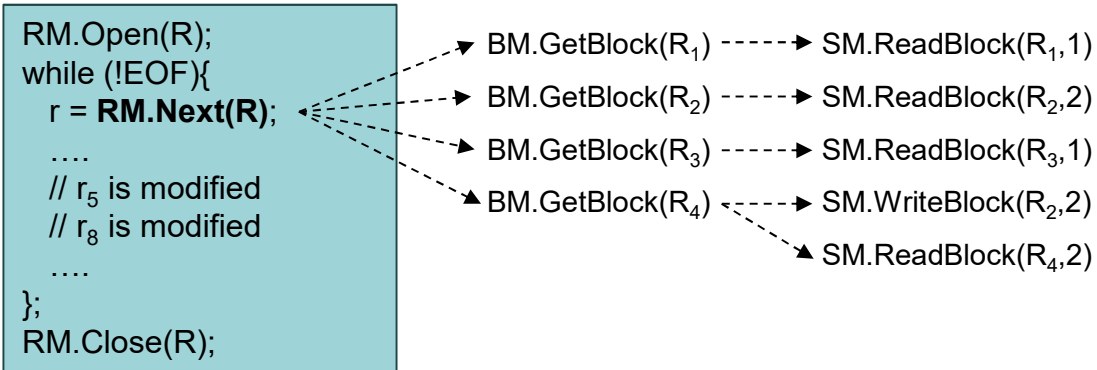
13.1 The Memory Hierarchy	557
13.1.1 The Memory Hierarchy	557
13.1.2 Transfer of Data Between Levels	560
13.1.3 Volatile and Nonvolatile Storage	560
13.1.4 Virtual Memory	560
13.2 Disks	562
13.2.1 Mechanics of Disks	562
13.2.2 The Disk Controller	564
13.2.3 Disk Access Characteristics	564

Data access



Processing of relation R

Query executor



Buffers in RAM

