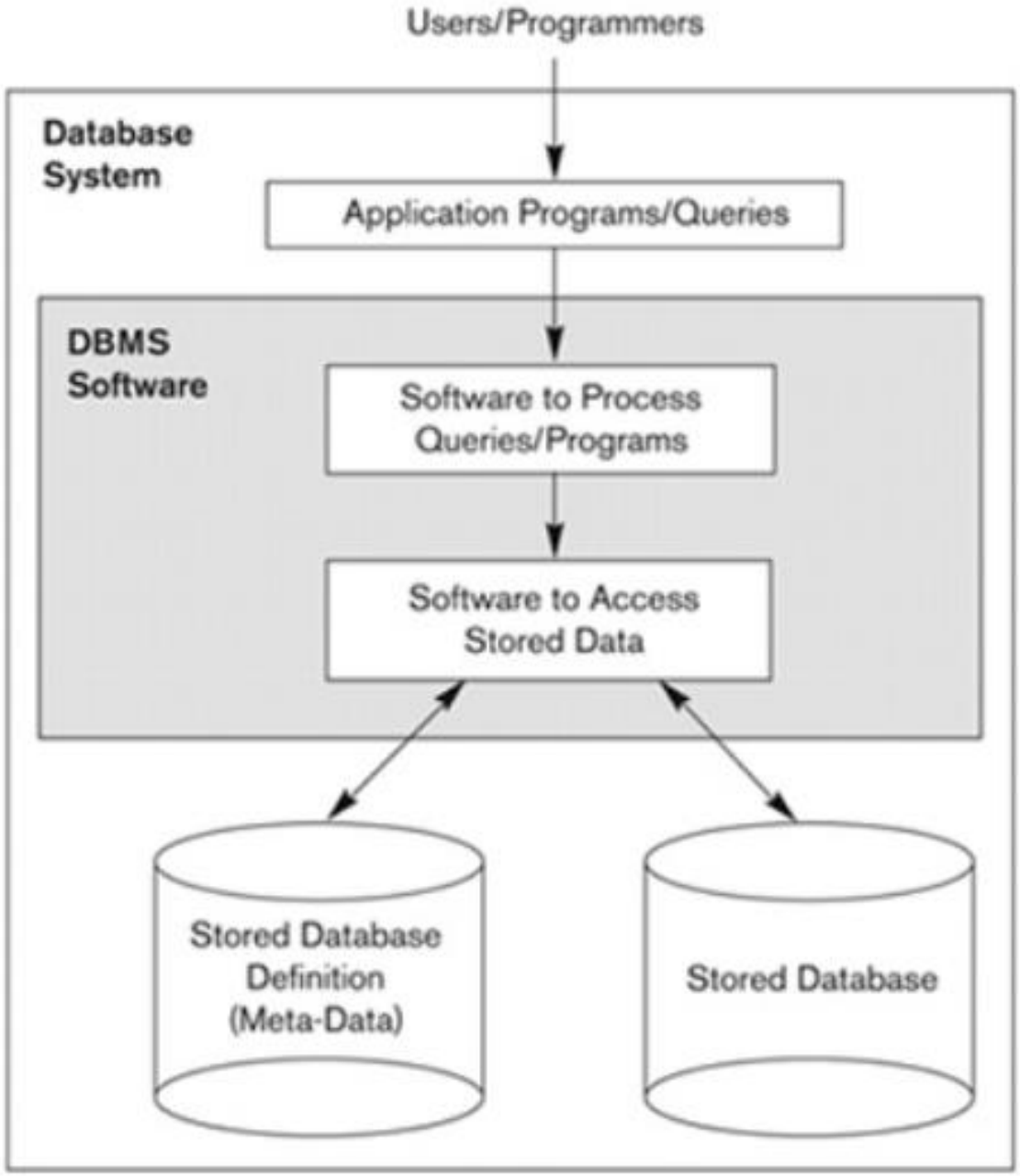
Relational Database Management System (RDBMS)

240110102

Simplified database system environment



Database Environment

- Database environment is a collective system of components that comprises and regulate the group of data, management and use of data which consists of s/w, h/w, people, techniques of handling DB and data also.
- DBMS: A piece of s/w that manages databases and lets you create, edit and delete databases, their tables and their data.
- DBMS are of 3 type:
  - RDBMS (Relational Database Management System): Stores its data and metadata in tables and access them using SQL
  - OODBMS (Object-Oriented Database Management System): Represent the data in the form of objects and uses OQL
  - ORDBMS(Object-Relational Database Management System): Represents some part of the data in the form of tables and some other part of the data in the form of objects. It uses SQL3 which is a combination of SQL2 and OQL
- Example of RDBMS
  - Oracle, DB2, IBM, MySQL, Microsoft SQLServer and PostgreSQL, SAP, Sybase ASE, Informix etc…

Characteristics of Database Approach

- Self-describing nature of a database system (Metadata)
- Insulation between programs and data – Data Abstraction (Program data Independence)
- Supports multiple views of the data
- Sharing of data and multiuser transaction processing

Data Models

- Underlying the structure of a database is the data model
- It is a collection of conceptual tools for describing data, data relationships, data semantics, and consistency constraints
- A data model provides a way to describe the design of a database at the physical, logical, and view levels
- There are a number of different data models that can be classified into four different categories
  - (Network data model and the Hierarchical data model – are now obsoleted model, only old database code uses it)
  - Relational model
  - ER Model
  - Object Based Data Model
  - Semi structured Data Model

Relational Model

- The relational model uses a collection of tables to represent both data and the relationships among those data.
- Each table has multiple columns, and each column has a unique name.
- Tables are also known as relations.
- The relational model is an example of a record-based model.
- Record-based models are so named because the database is structured in fixed-format records of several types.
- Each table contains records of a particular type.
- Each record type defines a fixed number of fields, or attributes.
- The columns of the table correspond to the attributes of the record type.
- The relational data model is the most widely used data model, and a vast majority of current database systems are based on the relational model

Entity-Relationship Model

- The entity-relationship (E-R) data model uses a collection of basic objects, called entities, and relationships among these objects.
- An entity is a “thing” or “object” in the real world that is distinguishable from other objects.
- The entity-relationship model is widely used in database design

Data dictionary

- The data dictionary is considered to be a special type of table that can only be accessed and updated by the database system itself (not a regular user).
- The database system consults the data dictionary before reading or modifying actual data
- Data dictionary contains Meta Data about the structure of the database, i.e. the schema of the database.
- A relational database system needs to maintain data about the relations, such as the schema of the relations. In general, such “data about data” is referred to as metadata
- Data dictionary is also called as System Catalog
- The data dictionary is often stored in a nonnormalized form to achieve fast access

Information stored in Data Dictionary are:

- Names of the relations.
- Names of the attributes of each relation.
- Domains and lengths of attributes.
- Names of views defined on the database, and definitions of those views.
- Integrity constraints (for example, key constraints).
- Names of authorized users.
- Authorization and accounting information about users.
- Passwords or other information used to authenticate users.
- Number of tuples in each relation.
- Method of storage for each relation
- If relations are stored in operating system files, the dictionary would note the names of the file (or files) containing each relation.
- If the database stores all relations in a single file, the dictionary may note the blocks containing records of each relation in a data structure such as a linked list.
- Name of the index.
- Name of the relation being indexed.
- Attributes on which the index is defined.
- Type of index formed.

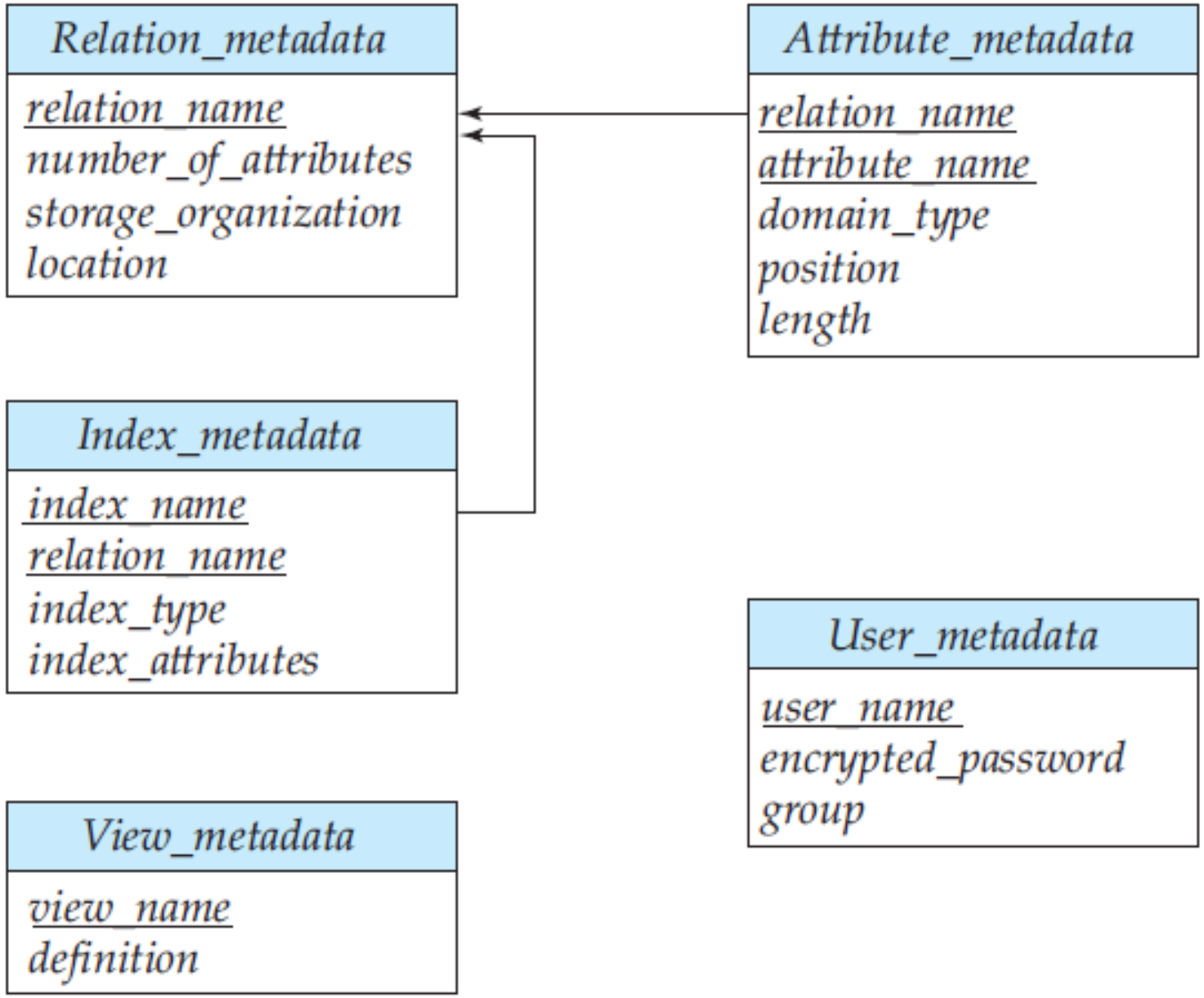


Figure 10.16 Relational schema representing system metadata.

Types of Data Dictionary

- There are two types of data dictionaries: active and passive
- Active data dictionary:
  - Any changes to the database object structure via DDLs will have to be reflected in the data dictionary. But updating the data dictionary tables for the changes are responsibility of database in which the data dictionary exists.
  - Active data dictionary is created in the same database, so the DBMS software will automatically update the data dictionary. Hence there will not be any mismatch between the actual structure and the data dictionary details.
  - The modification is an automatic task and most RDBMS has active data dictionary. It is also known as integrated data dictionary.
- Passive data dictionary:
  - Data dictionary created separately from the current database as entirely new database to store only data dictionary's information. Sometimes it is stored as xml, excels or in any other file format. In such case, an effort is required to keep data dictionary in sync with the database objects. This kind of data dictionary is called passive data dictionary. In this case, there is a chance of mismatch with the database objects and the data dictionary. This kind of DD has to be handled with utmost care.
  - This data dictionary is not a choice for many users as it is not easy to maintain
  - Managed by the users and is modified manually when the database structure change. Also known as non-integrated data dictionary.