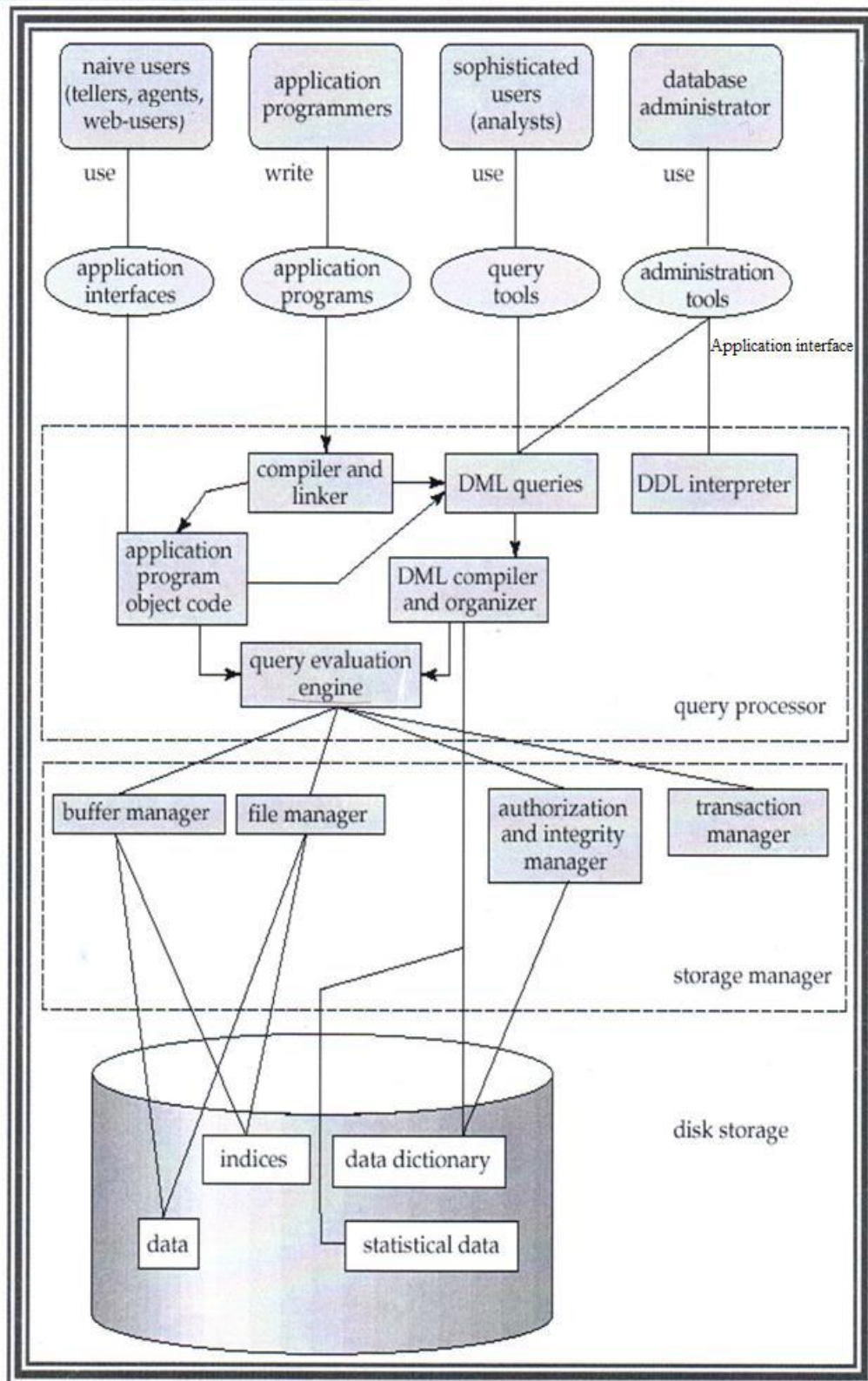
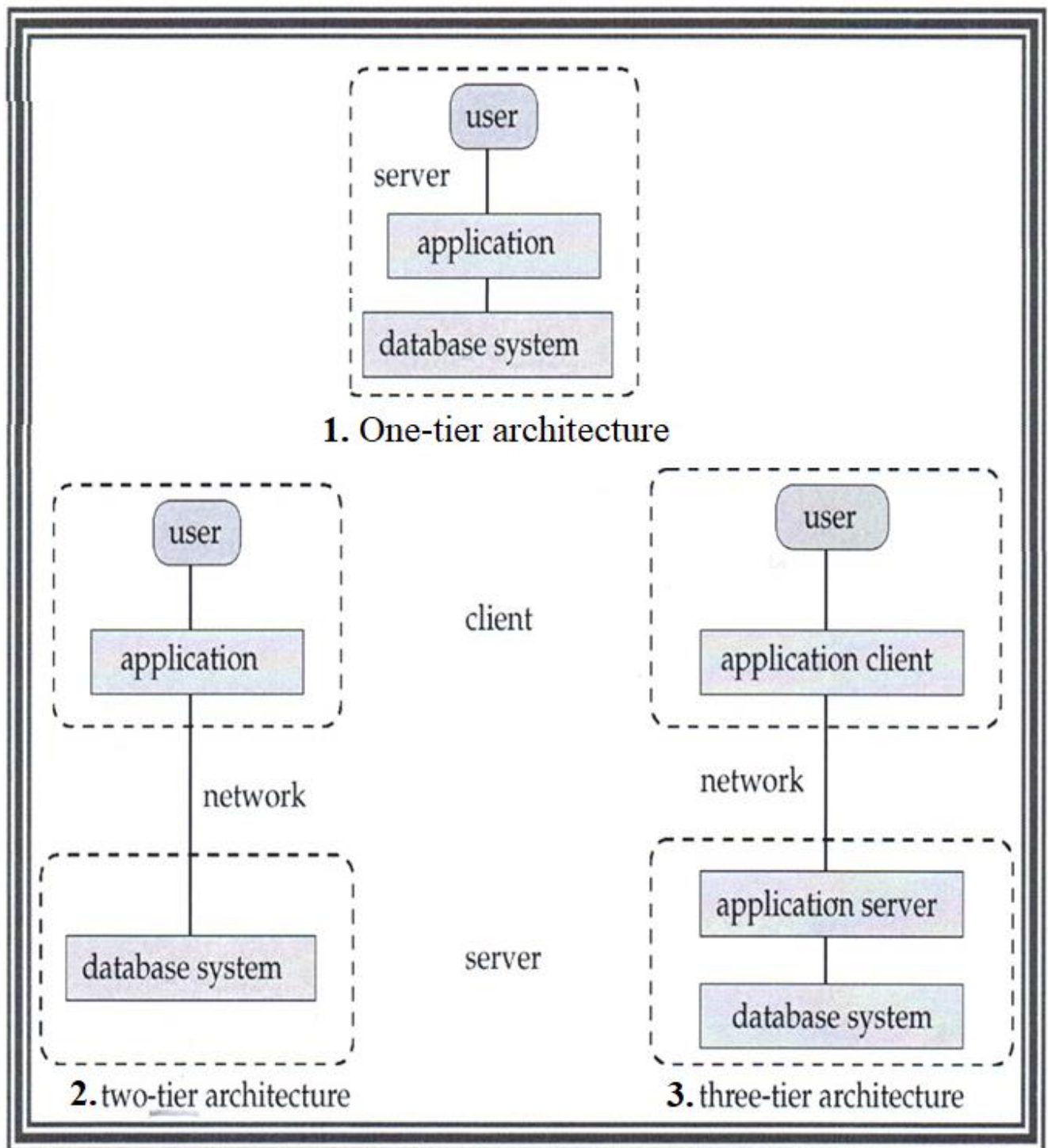


Second Lecture

Overall System Structure



Application Architectures



1. One-tier architecture: E.g. user application using a DBMS to access a database on one server.
2. Two-tier architecture: E.g. client programs using ODBC/JDBC to communicate with a database.
3. Three-tier architecture: E.g. web-based applications, and applications built using “middleware”.

Database Types

Most modern information systems utilize database technologies, either database management systems or data warehouse systems, for data storage and retrieval. Recall that a database is a collection of logically related data, designed to meet the information needs of multiple users in an organization. The relationship between files in a database is due to relationship. Identified in the conceptual and logical data models. The relationships imply access paths between data. Each type of database technology allows different types of access paths. So, the process of choosing the appropriate type of DBMS or data warehousing technology is one of matching the needed access path, with the capabilities of the database technology.

1. Hierarchical Database System

In this type, files are arranged in a top-down structure that resembles a tree or genealogy chart. Data are related in a nested, one-to-many set of relationships. The top file is called the root, the bottom files are called leaves, and intermediate files have one parent, or owner, file and one or several children files. Among the oldest of the database architectures, many hierarchical databases exist in larger organizations today. This technology is best applied when the conceptual data model also resembles a tree and when most data access begins with the same (root) file. Hierarchical database technology is used for high-volume transaction processing and MIS applications. Few new databases are developed with hierarchical DBMSs since newer applications tend to have broader needs than simply transaction processing or summarization of transaction data.

2. Network Database System

In this type, each file may be associated with an arbitrary number of files. Although very flexible because any, relationships can be implemented (a hierarchy is a special case of a network), the form of implementation, usually using pointers; between related records in different files, creates significant overhead in storage space and maintenance time. Typical k., network model systems support only one-to-many relationships along each arc in the network, but some support many-to-many relationships. Network model systems are still popular on powerful mainframes and for high-volume transaction processing applications. Since the database designer has such detailed control over data organizations, it is possible to design highly optimized databases with network systems. For example, each record type (a record type is shown by each box in the network) can be organized using hashing algorithms or located-near. another related record type-an early form of clustering. Network systems support a wider variety of processing requirements than do hierarchical database systems, but network systems still require significant programming and database design knowledge and time, and hence are used primarily in those organizations with significant expertise with such technologies.

3. Relational Database System

The most common database system for new systems defines simple tables for each relation and many-to-many relationships. Cross-reference keys link the tables together, representing the relationships between entities. Primary and secondary key indexes provide rapid access to data based upon qualifications. Most new applications are built using relational DBMS's and many relational DBMS products exist.

4. Object-Oriented Database System

In this type, attributes and methods that operate on those attributes are encapsulated in structures called object classes. Relationships between object classes are shown, in part, by nesting or encapsulating one object class within another object classes are defined from more general object classes.

A major advantage of this data model is that complex data types like graphics, video, and sound are supported as easily as simpler data types.

This is the newest DBMS technology and larger organizations are gaining experience with it by selectively using it when complex data or event-driven programming is appropriate for the application.

5. Multidimensional Database System

This database system is used in data warehousing applications. Two ways of viewing this model exist. The first views data as a multidimensional table in which each cell contains one or more simple attributes and the dimensions are ways to categorize the raw data. These categories, or dimensions, are the factors on which users want to summarize or segment the data, such as time period, geography, lines of business, or people. A cell contains data relevant to the intersection of all its dimension values. For example, a cell might hold the number-of-units-sold attribute for a given time period, location, line of business, and salesperson. The second (equivalent) view called a star schema. At the center is a fact table, equivalent to the cell in the multidimensional view. This table contains all the raw attributes and a composite key made up of the primary keys of all the surrounding dimension tables. The surrounding dimension tables define each of the ways to categorize data, such as all the description data about each salesperson.

6. Distributed Database System

This database system is used in distributed system. A distributed database is a collection of data that is spread across multiple physical locations, which can be within the same network or across geographically dispersed sites. These locations are interconnected, and the distributed database appears to users as a single unified database system.