Course Name:
Database Management Systems

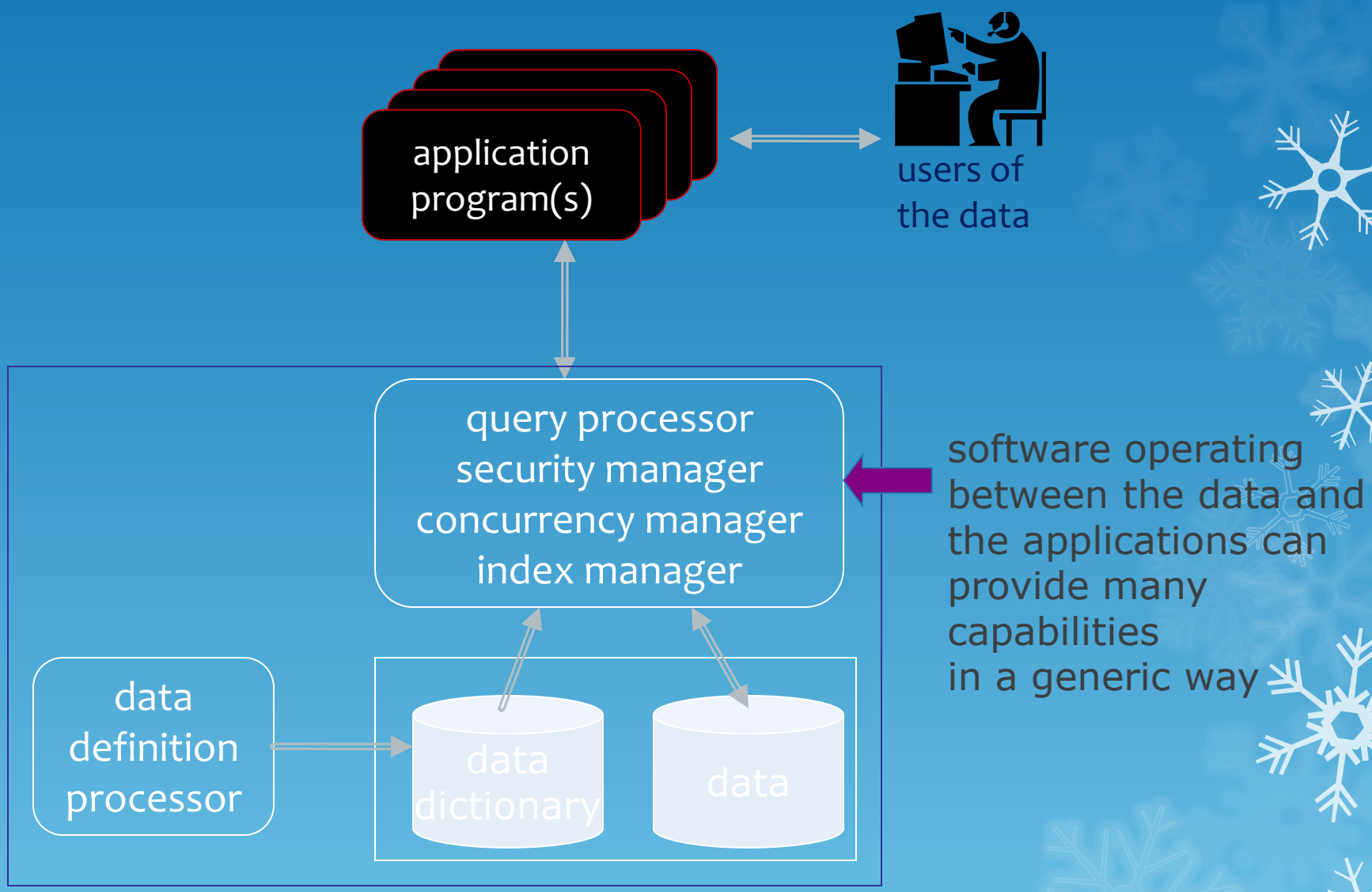
Lecture 2
Topics to be covered

- **Advantages of using a DBMS over File Systems**
- **Responsibility of Database Administrator**

Advantages of Using a DBMS

- Anything you can do with a DBMS, you can do with a file system, a network and a heap of C code
- So why spend the money to buy a DBMS?
  - there is a well defined collection of capabilities common to a certain class of applications
  - for applications in this class, the DBMS already has these capabilities and probably does them better than you could with home-brewed code

Advantages of Using a DBMS



software operating between the data and the applications can provide many capabilities in a generic way

Persistence

- A DBMS provides persistent objects, types and data structures
  - ***persistent*** = having a lifetime longer than the programs that use the data
  - any information that fits the data model of a particular DBMS can be made persistent with little effort
  - ***data model*** = concepts that can be used to describe the data

Concurrency

- A DBMS supports access by concurrent users
  - *concurrent* = happening at the same time
  - concurrent access, particularly writes (data changes), can result in inconsistent states
    (even when the individual operations are correct)
  - the DBMS can check the actual operations of concurrent users, to prevent activity that will lead to inconsistent states

Access Control

- A DBMS can restrict access to authorized users
  - security policies often require control that is more fine-grained than that provided by a file system
  - since the DBMS understands the data structure, it can enforce fairly sophisticated and detailed security policies
    - on subsets of the data
    - on subsets of the available operations

Redundancy Control

- A DBMS can assist in controlling redundancy
  - *redundancy* = multiple copies of the same data
  - with file storage, it's often convenient to store multiple copies of the same data, so that it's "local" to other data and applications
  - this can cause many problems:
    - wasted disk space
    - inconsistencies
    - need to enter the data multiple times

Complex Semantics

- A DBMS supports representation of complex relationships and integrity constraints
  - the semantics (meaning) of an application often includes many relationships and rules about the relative values of subsets of the data
  - these further restrict the possible instances of the database
  - relationships and constraints can be defined as part of the schema

Backup and Recovery

- A DBMS can provide backup and recovery
  - ***backup*** = snapshots of the data particular times
  - ***recovery*** = restoring the data to a consistent state after a system crash
  - the higher level semantics (relationships and constraints) can make it difficult to restore a consistent state
  - ***transaction analysis*** can allow a DBMS to reconstruct a consistent state from a number of backups

Views and Interfaces

- A DBMS can support multiple user interfaces and user views
  - since the DBMS provides a well-defined data model and a persistent data dictionary, many different interfaces can be developed to access the same data
  - data independence ensures that these UIs will not be made invalid by most changes to the data
  - new user views can be supported as new schemas defined against the conceptual schema

Advantages of Using a DBMS

- persistent objects, types and data structures
- control of concurrent users
- controlling of redundancy
- restricting access (security)
- representation of complex relationships and integrity constraints
- backup and recovery
- multiple user interfaces and user views

Database Users

Users are differentiated by the way they expect to interact with the system

- **Application programmers** – interact with system through DML calls
- **Sophisticated users** – form requests in a database query language
- **Specialized users** – write specialized database applications that do not fit into the traditional data processing framework
- **Naïve users** – invoke one of the permanent application programs that have been written previously
  - Examples, people accessing database over the web, bank tellers, clerical staff

Database Administrator

- Coordinates all the activities of the database system
  - has a good understanding of the enterprise's information resources and needs.

- Database administrator's duties include:
  - Storage structure and access method definition
  - Schema and physical organization modification
  - Granting users authority to access the database
  - Backing up data
  - Monitoring performance and responding to changes
    - Database tuning

Database Architecture

The architecture of a database systems is greatly influenced by the underlying computer system on which the database is running:

- Centralized
- Client-server
- Parallel (multiple processors and disks)
- Distributed