

CSC 742
Database Management Systems

Topic #2:
Concepts and Architecture

Spring 2002

CSC 742: DBMS by Dr. Peng Ning

1

Concepts: 1

- *Data model* (really *metamodel*):
 - ◆ A language for describing a database, new one every few years
 - ◆ High-level, low-level, logical
- *Schema*:
 - ◆ A description of the database, changes as often as every few months
- *Database*:
 - ◆ The data organized to match the schema, may change continually

Spring 2002

CSC 742: DBMS by Dr. Peng Ning

2

Concepts: 2

- *Extension:*
 - ◆ what the database contains at a moment
- *Intension:*
 - ◆ what the database is designed to contain—schema plus constraints

Kinds of Metamodel

- Conceptual (close to user needs)
 - ◆ Entity-relationship
- Logical (close to representation)
 - ◆ Relational, O-O, O-R
- Physical (close to file system)
 - ◆ Disk data structures, index structures, access path

Three-Schema Architecture

- *External* view: one per user class
- *Conceptual* schema: common basis of all external views
- *Internal* schema: description of physical layout

Data Independence

- Data independence
 - ◆ The capacity to change the schema at one level of a DBMS without having to change the schema at the next higher level.
- Mappings among adjacent levels provide data independence
 - ◆ *logical*: upper mapping
 - ◆ *physical*: lower mapping

Achtung!

- External views use conceptual metamodels (sometimes also logical)
- Conceptual schemas use conceptual and logical metamodels

DBMS Languages: 1

- Data definition (DDL)
 - ◆ sometimes split into
 - ◆ SDL (storage)
 - ◆ VDL (view)

DBMS Languages: 2

- Data manipulation (DML)
 - ◆ declarative:
 - ◆ independently usable
 - ◆ set-oriented
 - ◆ called a query language (includes updates, though)
 - ◆ procedural:
 - ◆ dependent on host language
 - ◆ record-oriented

DBMS Languages: 3

- Even declarative DMLs may serve as a data sublanguage by being embedded into a host language.

User Interfaces

- *Terminal-based
- *Forms-based
- Browsing: to explore a database
- Special-purpose: for parametric users
- Menu-based
- Graphical: schema-based query formulation
- For administrators
- (*: you can choose either for your project)

Components: Bare Essentials

- Disk
- Stored data manager
- Run-time processor



Components for Data Definition

- DDL compiler
- System catalog



Components for Access

- For casual users
 - ◆ Query compiler
- For application programs
 - ◆ Pre-compiler
 - ◆ DML compiler

Components for Sharing

- Concurrency control
- Backup
- Recovery
- Communication software
 - ◆ client-server
 - ◆ middleware

Utilities

- Load
- Convert formats
- Backup
- Reorganize files indexes
- Monitor performance
- Data dictionary: catalog plus information on requirements and designs
 - ◆ passive: usable by humans
 - ◆ active: usable by humans and programs