Overview of Database Management

Olaf Hartig

David R. Cheriton School of Computer Science
University of Waterloo

CS 640
Principles of Database Management and Use
Winter 2013

These slides are based on a slide set provided by Prof. M. Tamer Özsu.

Notes

What is "Data"?

- ANSI definition of data:
  1. A representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means.
  2. Any representation such as characters or analog quantities to which meaning is or might be assigned. Generally, we perform operations on data or data items to supply some information about an entity.
- Volatile vs persistent data
  - Our concern is primarily with persistent data

Notes

Representation of Data

Data Model Formalism that defines how data is structured and how it can be accessed and manipulated

Examples:

- Hierarchical; network
- Relational
- Object-oriented (more recently object-relational)

Schema Definition of how a particular collection of data is organized, using a given data model

Instance A particular collection of data

- Instances conform to a schema.
- A schema can have many instances.

Notes

Representation of Data (Example)

- Data model: Relational
- Schema:
  - EMP(ENO, ENAME, TITLE)
  - PROJ(PNO, PNAME, BUDGET)
  - WORKS(ENO, PNO, RESP, DUR)
- Instance:

EMP

ENO	ENAME	TITLE
E1	J. Doe	Elect. Eng
E2	M. Smith	Syst. Anal.
E3	A. Lee	Mech. Eng.
E4	J. Miller	Programmer
E5	B. Casey	Syst. Anal.
E6	L. Chu	Elect. Eng.
E7	R. Davis	Mech. Eng.
E8	J. Jones	Syst. Anal.

PROJ

PNO	PNAME	BUDGET
P1	Instrumentation	150000
P2	Database Develop.	135000
P3	CAD/CAM	250000

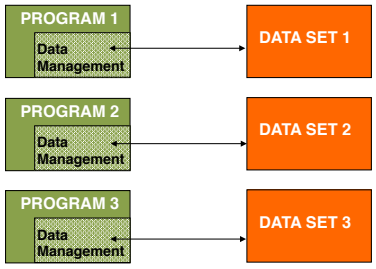
WORKS

ENO	PNO	RESP	DUR
E1	P1	Manager	12
E2	P1	Analyst	24
E2	P2	Analyst	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36
E8	P3	Manager	40

Notes

Early Data Management – Ancient History

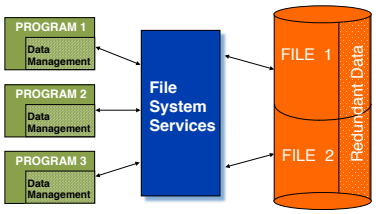
- Data are not stored on disk
- One data set per program. High data redundancy



Notes

File Processing – More Recent History

- Data are stored in files with interface between programs and files.
- Various access methods exist (e.g., sequential, indexed, random).
- One file corresponds to one or several programs.

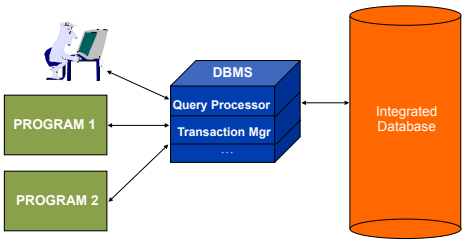


Notes

Database Approach

Definition (Database)

A *large* and *persistent* collection of (more-or-less similar) pieces of information organized in a way that facilitates efficient *retrieval* and *modification*.



Notes

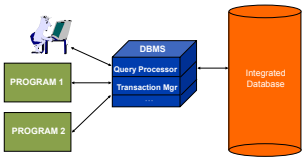
Database Approach (cont'd)

Definition (Database)

A *large* and *persistent* collection of (more-or-less similar) pieces of information organized in a way that facilitates efficient *retrieval* and *modification*.

Definition (Database Management System (DBMS))

A software system designed to store and manage databases.



Notes

Database Management System

Idea

Abstracts common functions and creates a uniform well defined interface for applications accessing data.

1. Data model
   all data stored in a well defined way
2. Access control
   only authorized people get to see/modify it
3. Concurrency control
   multiple concurrent applications access data
4. Database recovery
   nothing gets accidentally lost
5. Database maintenance
   structures are modified to accommodate evolving needs

Notes

Data Independence

Idea

Applications do not access data directly but, rather through an abstract data model *provided by the DBMS.*

Two kinds of data independence:

- Physical: applications immune to changes in storage structures
- Logical: applications immune to changes in data organization

Note

One of the most important reasons to use a DBMS!

Notes

Three Levels of Abstraction

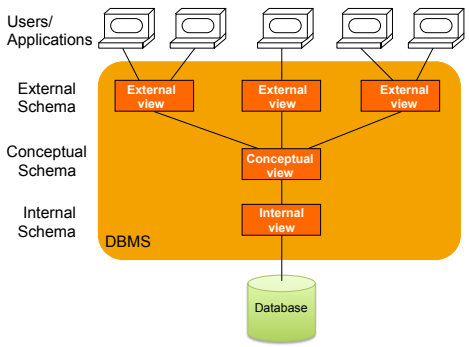
External Schema (View) What an application program or user sees (may differ for different users of the same database)

Conceptual Schema Description of the logical structure of *all* data in the database

Physical Schema Description of physical aspects of how data is structured (selection of files, devices, storage structures, etc.)

Notes

Three Levels of Abstraction (cont'd)



Notes

Types of Database Users

End user:

- Accesses the database indirectly through forms or other query-generating applications, or
- Generates ad-hoc queries using a dedicated language.

Application developer:

- Designs and implements applications that access the database.

Database administrator (DBA):

- Manages conceptual schema.
- Assists with application view integration.
- Monitors and tunes DBMS performance.
- Defines internal schema.
- Loads and reformats database.
- Is responsible for security and reliability.

Notes

Primary Interface to a DBMS

Data Manipulation Language (DML) for specifying queries and modifying the data

- **navigational** (procedural)
- **non-navigational** (declarative)

Data Definition Language (DDL) for specifying schemas

- may have different DDLs for external schema, conceptual schema, internal schema
- information is stored in the **data dictionary**, or **catalog**

Notes

Summary

Using a DBMS to manage data helps:

- to remove common code from applications
- to provide uniform access to data
- to guarantee data integrity
- to manage concurrent access
- to protect against system failure
- to set access policies for data

Notes