پوهنتون کابل

پوهنحی کمپیوترساینس

Introduction to Database and Data Models

Lectures 01-02

تهیه کننده : پوهنیار محمد شعیب "زرین خیل"
سال : 1389

Introduction to Database and Data Models

01

By: M Shuaib Zarinkhail

2010

Course Overview

- Syllabus:
  - Instructors
  - Text books
  - Grading
  - Schedule

Instructors

- Mohammad Shuaib Zarinkhail
  - Assistant Professor @ Computer Science Faculty / Kabul University
- Ogai Ahmadi
  - Assistant Professor @ Computer Science Faculty / Kabul University

Teaching Materials

- Text note and Presentations
- Prepared from the following books:
  - An Introduction to Database Systems. Eight Edition. By: C. J. Date. 2004
  - Database Concepts. By: Kroenke. 2002
  - DATABASE SYSTEMS: Design, Implementation, and management.

    By: P. Rob and C. Coronel. 1997

Grading

Activity	**KU-Scores**	**Standard-Scores**
Lab Works	10%	20%
Projects	20%	30%
Midterm	10%	20%
Final	60%	30%
Total	**100%**	**100%**

Schedule

- Lecture (Room #)
  - Saturdays
    - Second hour
      - (9:40 to 11:10)
- Lecture and Practice (Room #)
  - Mondays
    - Second and third hours
      - (9:40 to 11:10 – 11:20 to 12:50)

Contents (DB-1)

- The Data Modeling Process
- Basic Relational Concepts
- The Process of Normalization
- Labs in MS-Access
- Guidelines for Mapping a Data Model Into a Relational Database

Contents (DB-2)

- Relational Algebra
  - Relational Algebra Operations
- SQL (Structured Query Language)
  - Data Definition Language (SQL-DDL)
  - Data Manipulating Language (SQL-DML)
  - SQL Internal Functions
  - SQL For Relational Queries

Introduction to Databases

Topics:

- What is a database?
- Alternative to databases
- Database advantages
- Database disadvantages
- History

What is a database? (definition)

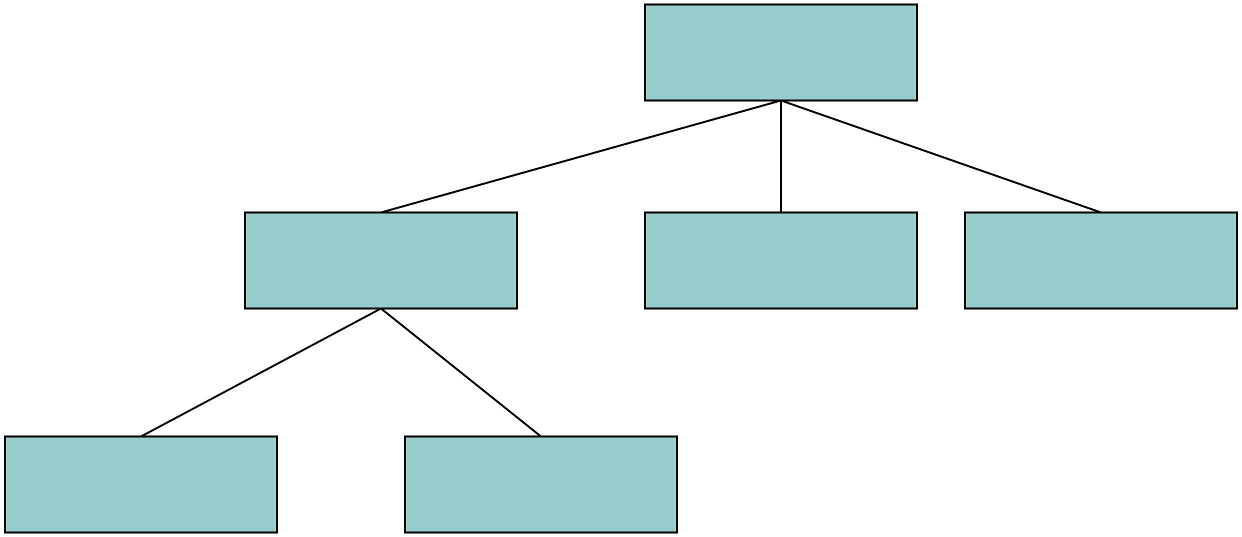
- A database is a self describing collection of integrated records. Kroenke 2002
- An organized collection of logically related data. McFadden, et al. 1999
- A shared collection of logically related data, and a description of this data, designed to meet the information needs of an organization. Connolly, Begg 2002

Database Models

- Hierarchical – original form
- Network – later variant
- Relational – most common, subject of this course
- Object oriented – recent development

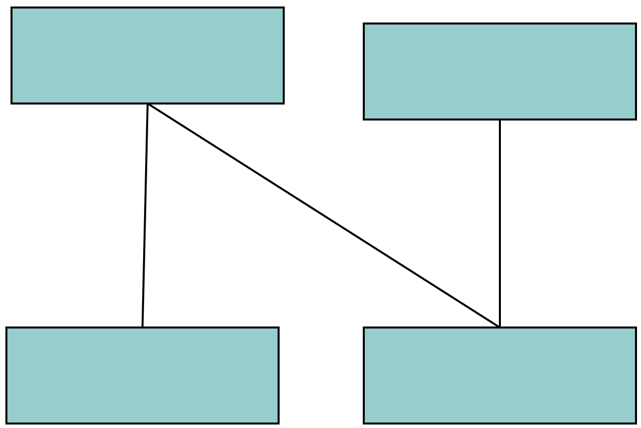
Hierarchical Model

- Hierarchical Model
  - Very limited
  - Very slow if not following hierarchy



Network Model

- Less restrictive than Hierarchical
- Pure network model is still not ideal



Types of Data

- We can store any of the following in a DB:
  - Texts
  - Numbers
  - Date / Time
  - Graphics and images
  - Sounds and videos
  - Programs and other objects

Metadata

- Metadata: data about data
- Metadata includes:
  - Data definitions
  - Data structures
  - Constraints
  - Rules

Userdata

- Data entered to a database by users
- Userdata can be
  - Updated
  - Deleted
  - Retrieved by users

Database Examples

- A DRIVER LICENCE COMPANY:
  - Stores data about registered vehicles and drivers
  - Stores images for drivers' licenses
  - Keeps records of driver infractions (points)
  - Knows about insurance companies
  - Registers repair shops
  - Keeps Information about car check-ups

Database Examples

- A SHOPPERS CLUB:
  - Keeps records of transactions
  - Knows the shopper’s demographics
  - Knows which stores the shopper commonly use
  - Authorizes check cashing
  - Also works at other stores

Database Types

- Personal
  - Can be created by MS-Access
- Departmental / Workgroup
  - Can be created by MySQL, SQL Server
- Enterprise
  - Can be created by Oracle, Sybase, Informix, SQL Server

Alternative to Databases

File Systems

- Older technology
- Many legacy systems still in use
- Often seen as a desirable alternative to databases

File System Disadvantages

- Application program dependency
- Data duplication
- Separated and isolated data
- Long development times
- Increased maintenance requirements

Application Program Dependency

- Each file has a specific format defined when created
  - Programs need to follow this format when getting or adding data
- Changes to the file format require program changes
- A file change could affect multiple programs

Data Duplication

- Separate groups may have their own data files
- Data could appear in multiple files
- Requires more storage space
- Changes may not get to all files

Separated and Isolated Data

- Very difficult to resolve differing formats to share information
- Could require creation of new programs to share the data
- Political issues - who owns the data?
- How to deal with incompatible data between files?

Long Development times

- Most file programs are custom-built
- Most organizations start from scratch with each new program
- Not the most exciting work

Increased Maintenance Requirements

- Up to 80% of a department’s budget may go to maintaining old file based programs:
  - New Reports
  - Need to add/remove data elements
  - Need to handle new values
  - Error correction

Introduction to Database and Data Models

02

By: M Shuaib Zarinkhail

2010

Database Advantages

- Program/data independence
- Reduced data duplication
- Data consistency
- Data sharing
- Faster application development

Database Advantages

- Standards enforcement
- Better data quality
- Improved accessibility
- Less maintenance

Program / Data Independence

- Because data descriptions are separate from the data, we can write format independent programs
  - Adding new attributes to a database will not affect a program
- Programs that refer to the metadata directly may not require updates as the database changes

Reduced Data Duplication

- A properly designed database should only store one item of data in one place

- Changes to an item’s value will be seen by all users

Data Consistency

- This feature comes out of reduction in redundancy

- While data appears in one place, there should be no consistency problems

Data Sharing

- Most databases allow multiple users to access data
  - Sharing can be controlled by the database's security features
  - Format considerations are no longer a concern!

Faster Application Development

- Faster development can occur because:
  - The basic database design has already been done and new applications may be adding some features, but no changing of the entire database
  - Existing code can be reused to access the database
  - Reports and forms are often provided as part of the database

Standards Enforcement

- Within a single database, corporate standards for data element naming, data types, supported values, ... can be enforced
- A standard set of definitions can be created and used by many projects

Better Data Quality

- Constraints stored in the database can prevent bad data from being entered

- Standard methods of checking data can be created and used when loading data

Improved Accessibility

- End users can potentially get their own data using Structured Query Language (SQL)
- SQL can be learned quickly (We will cover it in Fall semester)
- SQL is a standard language used in many databases

Less Maintenance

- SQL is a standard language used in many databases
  - This goes back to “Program-Data Independence” that we saw earlier

Comparison (Data Separation & Isolation)

File Processing

- Data stored in separate files
- Special programs must be written to match records and extract data
- Complicated and time consuming

Database Processing

- All application data is stored together in the database
- The DBMS has algorithms to automatically relate and combine data

Comparison (Data Duplication)

File Processing

- Data items frequently often duplicated across records and files
- Difficult to update
  - Have problems with data integrity

Database Processing

- Minimal data duplication
  - Data items typically stored only once
  - Key fields often duplicated but can have special algorithms to maintain data integrity
    - Referential integrity

Comparison (Program / Data Independence)

File Processing

- If data file is modified, all programs that use that file must be updated
  - Time consuming (must change and test each program) and is error prone

Database Processing

- The format of the data is stored as part of database
- DBMS does all data access
- Application programs only need to identify data needed
  - Not affect by format changes

Comparison (File Compatibility)

File Processing

- May have to convert files to compatible formats

Database Processing

- DBMS does all data access – not application programs

Comparison (Ability to Represent the User's Perspective of Data)

File Processing

- Separate files do not have the data relationships that the user needs to get information that is useful for decision making

Database Processing

- Relationships between data items are physically stored as part of the database

Database Disadvantages

- Trained personnel
- Installation and management costs
- Conversion costs
- Backup and recovery
- Political problems

Trained Personnel

- Databases require specially trained personnel
  - DBAs (Database Administrators)
  - Data modelers
- Training is expensive
- Retraining as new features appear
- Limited availability of these people

Installation & Management Cost

- Software costs can be high
  - More than $1,000,000 in some cases
- More than $1,000,000 in some cases
- May need to upgrade hardware or OS
- May need to come up with new policies
  - User access
  - Privacy
  - Data control

Conversion Costs

- Taking old data from files or other sources is time consuming
- Some tools provided:
  - Loader programs
  - Direct conversion from other DBMSs
- Usually a custom solution is needed

Backup and Recovery

- Centralized data is now a corporate resource
- Need to protect against:
  - Attacks
  - Media failure
  - Hardware failure
- Need to back up data and have a way to retrieve it

Political Problems

Who owns the data?

- Who can change the data?
- Who can see the data?

History

1960’s:

- File processing systems
- Initial efforts at database systems in late 60’s

1970’s:

- Hierarchical and network databases
- Development of relational data model by E. F. Codd and others

History

1980's:

- Development of commercial databases by Oracle, Sybase and others
- Structured Query Language (SQL) developed

1990's:

- Rise of Client Server Computing
- Ability to store data types aside form text, numbers, …

History

1990's:

- Large storage options
- Object Oriented databases

2000's

- User defined data types
- Distributed databases

Database Technology Trends

	1960s to Mid-1970s	1970s to Mid-1980s	Late 1980s	Future
Data Model	Network Hierarchical	Relational	Semantic Object-oriented Logic	Merging data models, knowledge representation, and programming languages
Database Hardware	Mainframes	Mainframes Minis PCs	Faster PCs Workstations Database machines	Parallel processing Optical memories
User Interface	None Forms	Query languages - SQL, QUEL	Graphics Menus Query-by-forms	Natural language Speech input
Program Interface	Procedural	Embedded query language	4GL Logic programming	Integrated database and programming language
Presentation and display processing	Reports Processing data	Report generators Information and transaction processing	Business graphics Image output Knowledge processing	Generalized display managers Distributed knowledge processing

Recent Developments

- Object Oriented databases
- Client Server systems plus databases
- Distributed databases
- Data Warehouses

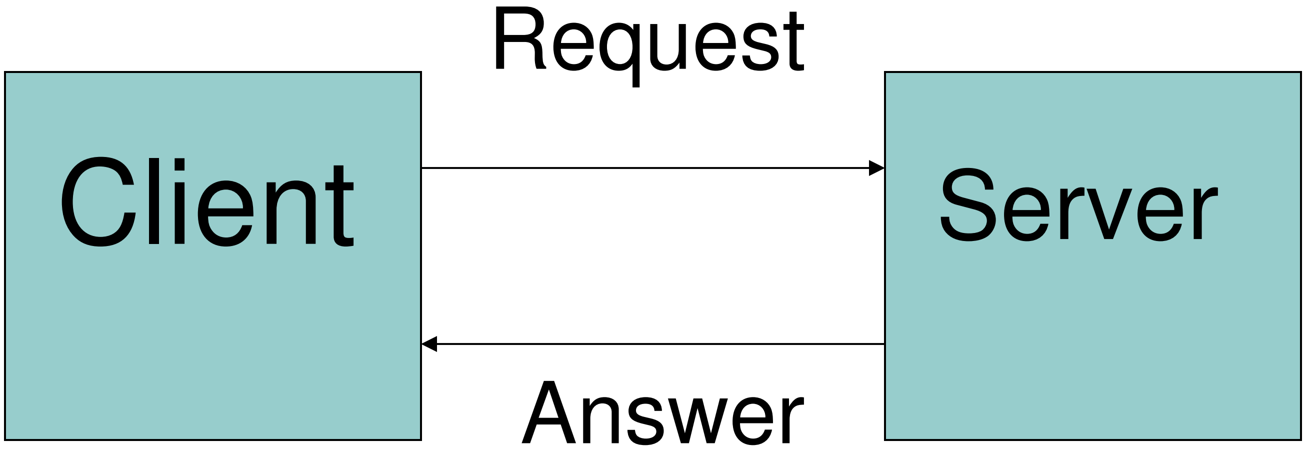
Object Oriented Databases

- Objects:
  - Store both data and methods
- Methods are instructions executed on data
- Can get the object from the database, look at the data or execute the methods

Client Server

- The rise of networking has led to this development
- Client requests services from another program or machine
- Server provides requested services
- Client and server can be on different machines or on the same machine

Client Server



Client Server

Database server:

- Is a machine that houses a database
- Clients request data via queries
- Server provides the requested data to its clients on the network

Distributed Databases

- Distributed database: “a database stored on two or more machines.” Kroenke,2002
- Types:
  - Replicated
  - Partitioned
- Still has some problems with consistency

Data Warehousing

- Storing enterprise wide data
- Example: A Shoppers Club
  - Used for decision making
  - Large scale relational database – terabytes of data
  - Often used third party tools to analyze the data