CS3402
Database Systems

Teaching Staff's Information

- **Dr. LIAO Jing** (Lecturer)
  - Email: jingliao@cityu.edu.hk
  - Phone: 3442-8619
  - Office: G6429

- **WANG Xinyu, QURESHI Umair Mujtaba, CUI Yufei** (Teaching Assistants)
  - Emails: xinywang-c@my.cityu.edu.hk (Mr. WANG),
    umqureshi2-c@my.cityu.edu.hk (Mr. QURESHI)
    yufeicui3-c@my.cityu.edu.hk (Mr. CUI)

Course Overview

- Course Format:
  - Regular lectures
    - Class timetable:
      - Tuesday 5:00pm - 6:50PM lectures (LT-8)
  - Tutorial and lab classes
      - One-hour tutorial **or** lab class every week
      - Patterns:

        w2-3, w7-9, w12-13: tutorial classes;

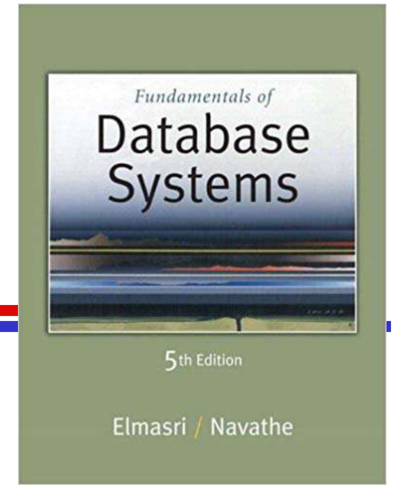
        w4-6, w10-11: lab classes (@MMW1411)

Assessment

- Coursework *-- 40%* :
  - Tutorial and lab attendance *-- 5%*
    - *Proportional to number of attendances*
    - *5% for at least 10 attendances*
  - Mid-term *-- 20%*
  - Homework assignments *-- 15%*

- Final examination *-- 60%*
  - *Get 30 out of 100 to pass*

Course Materials

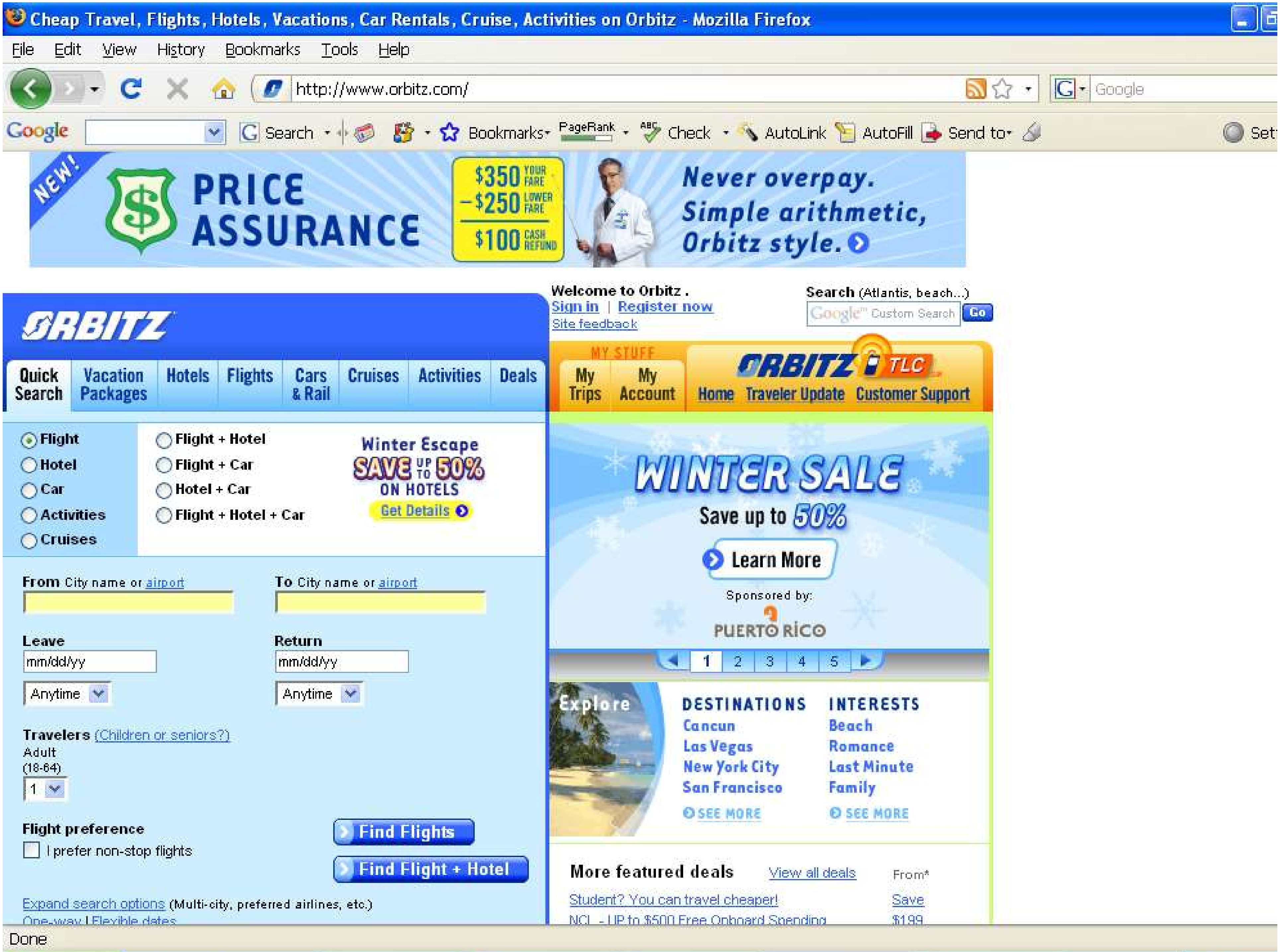


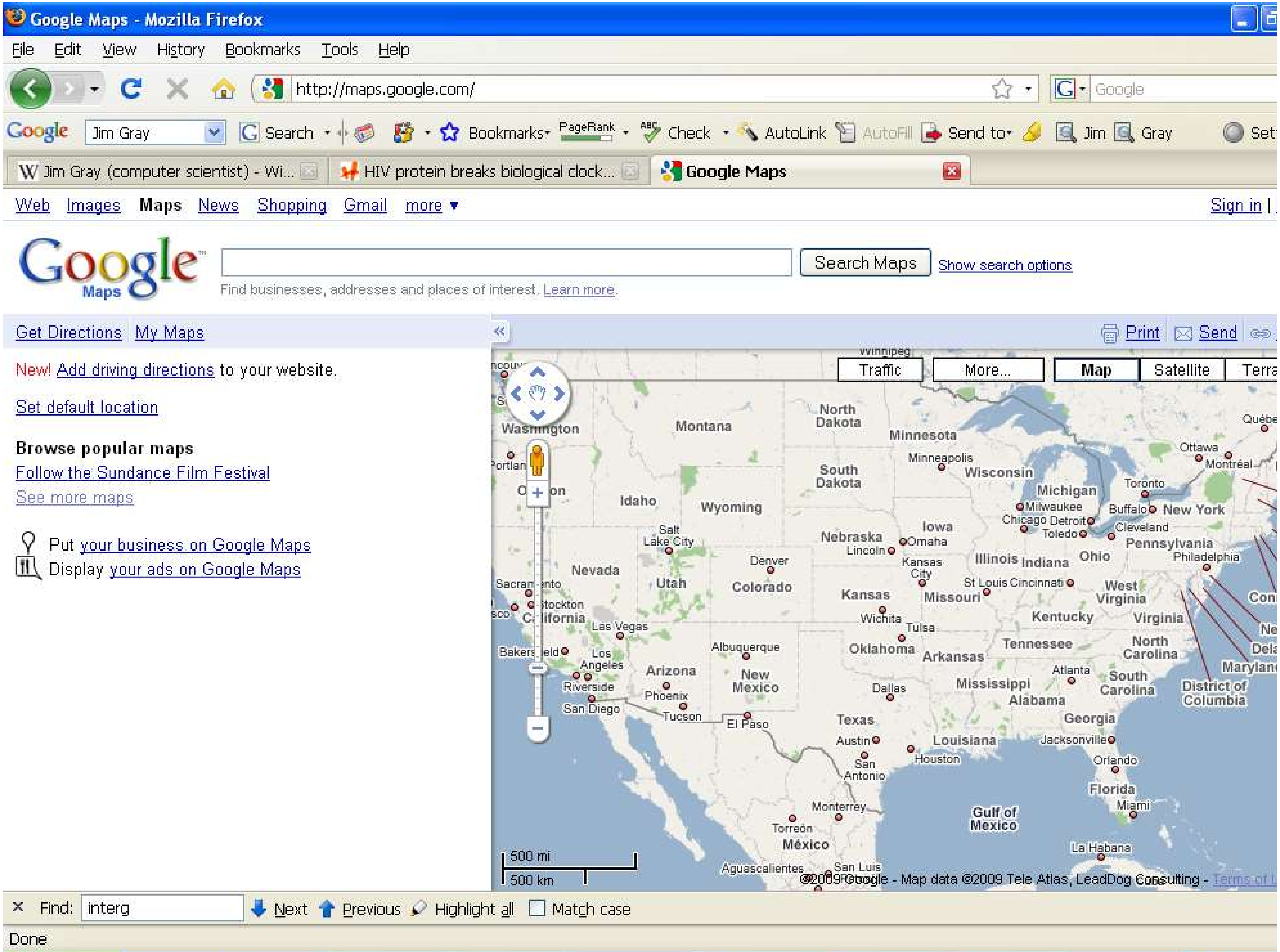
- Text books
  - "**Fundamentals of Database Systems**", 4th/5th edition (***or later***), by R. Elmasri, S.B. Navathe, Addison-Wesley.
  - "Database System Concepts", 5th edition (*or later*), by A. Silberschatz, H. Korth, S. Sudarshan, McGraw-Hill Companies Inc.
- Notations may vary in different books. Please stick to the ones used in this lecture notes!

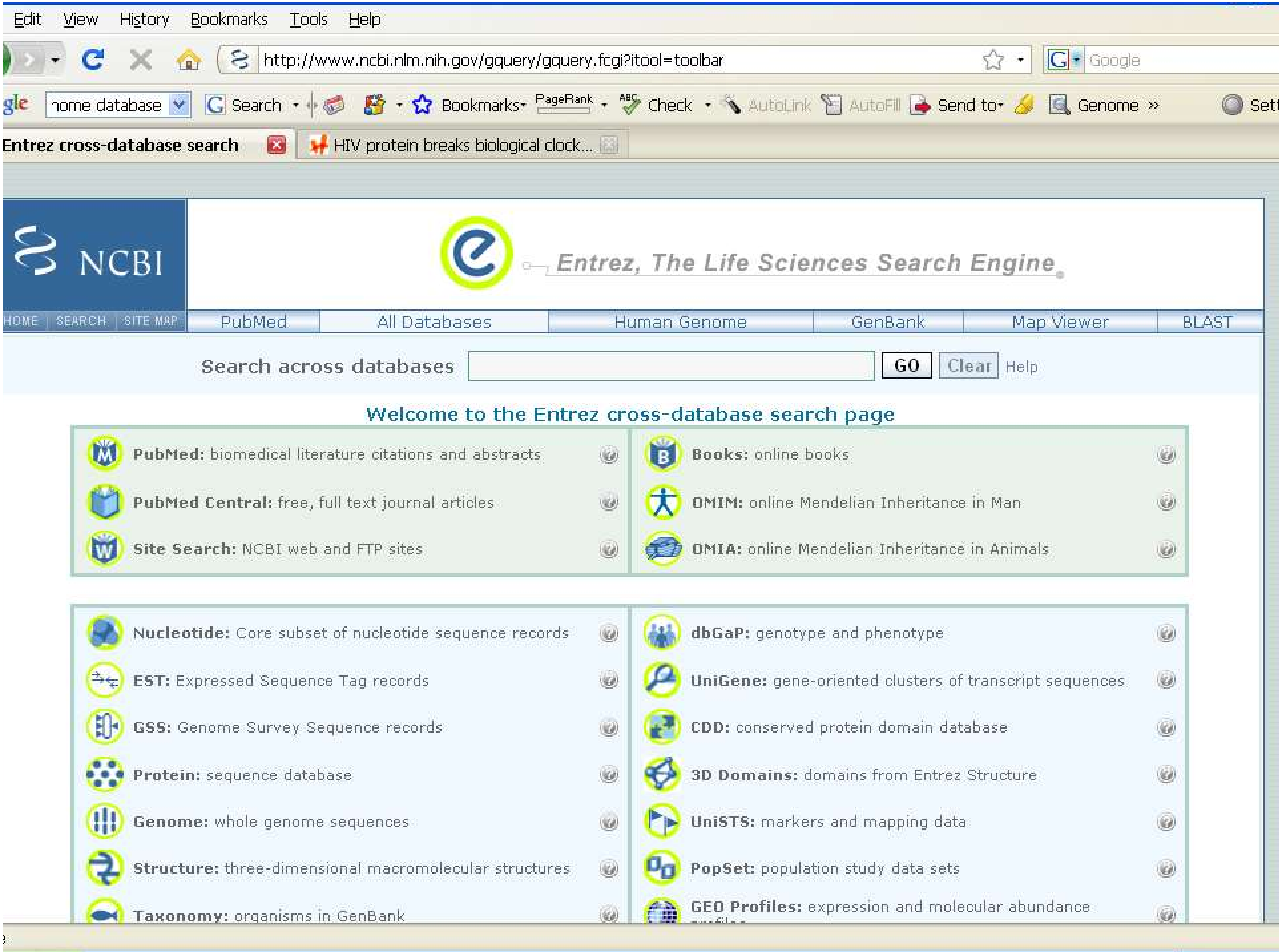
Databases is everywhere

- You cannot avoid it and it's everywhere!
- You can say it actually makes the current society and your life work!
- Banking/Credit card /Social Security Info…
- Online shopping/booking…


Amazon.com: Online Shopping for Electronics, Apparel, Computers, Books, DVDs & more - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www.amazon.com/
Google
Search
Bookmarks
PageRank
Check
AutoLink
AutoFill
Send to
amazon.com
Hello, Ruoming Jin. We have recommendations for you. (Not Ruoming?)
FREE 2-Day Shipping. No Minimum Purchase
Ruoming's Amazon.com
Today's Deals
Gifts & Wish Lists
Gift Cards
Your Account | Help
Shop All Departments
Search
All Departments
GO
Cart
Your Lists
Books
Movies, Music & Games
Digital Downloads
Computers & Office
Electronics
Home & Garden
Grocery, Health & Beauty
Toys, Kids & Baby
Apparel, Shoes & Jewelry
Sports & Outdoors
Tools, Auto & Industrial
100% More
amazonkindle
Books, Newspapers, and Blogs
We've doubled our selection since launch to over 200,000 titles—all available wirelessly in under 60 seconds:
• New York Times bestsellers are $9.99, unless otherwise marked
• Expanded newspaper selection, including The New York Times, Wall Street Journal, and Financial Times
• More magazines and blogs, from Newsweek to Time to Gawker
amazonkindle
Learn more
Amazon Daily BLOG
5 posts since yesterday
Posts for Ruoming
Treat Yourself
Combinatorial Optimization
From the reviews: What a book! Actually it is three books which come... Read more
$135.00 $107.51
See more in your Wish List
Get Your Motor Running
Explore Our New Motorcycle & ATV Store
Shop now at Amazon.com
Check This Out
Tax Central
Find tax software, books, and more
2009
Cast Your Vote
See our Top 10 New Year's Resolutions. Which one's for you?
Holiday Returns
Visit our self-service Online Returns Center.
For the Photography Fan
Zoom in on top
Frequently Bought with Items in Your Wish List
LOOK INSIDE!
Theory of Linear and Integer Programming
Paperback by Alexander
LOOK INSIDE!
Combinatorial Optimization...
Paperback by Christos H.
Approximation Algorithms Hardcover by Vijay V. Vazirani
Done


Cheap Travel, Flights, Hotels, Vacations, Car Rentals, Cruise, Activities on Orbitz - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www.orbitz.com/
Google
Search
Bookmarks
PageRank
Check
AutoLink
AutoFill
Send to
NEW!
PRICE ASSURANCE
$350 YOUR FARE
–$250 LOWER FARE
$100 CASH REFUND
Never overpay. Simple arithmetic, Orbitz style.
ORBITZ
Quick Search
Vacation Packages
Hotels
Flights
Cars & Rail
Cruises
Activities
Deals
Flight
Hotel
Car
Activities
Cruises
Flight + Hotel
Flight + Car
Hotel + Car
Flight + Hotel + Car
Winter Escape
SAVE UP TO 50% ON HOTELS
Get Details
From City name or airport
To City name or airport
Leave
mm/dd/yy
Anytime
Return
mm/dd/yy
Anytime
Travelers (Children or seniors?)
Adult (18-64)
1
Flight preference
I prefer non-stop flights
Find Flights
Find Flight + Hotel
Expand search options (Multi-city, preferred airlines, etc.)
One-way | Flexible dates
Welcome to Orbitz .
Sign in | Register now
Site feedback
Search (Atlantis, beach...)
Google Custom Search
Go
MY STUFF
My Trips
My Account
ORBITZ TLC
Home
Traveler Update
Customer Support
WINTER SALE
Save up to 50%
Learn More
Sponsored by:
PUERTO RICO
1 2 3 4 5
Explore
DESTINATIONS
Cancun
Las Vegas
New York City
San Francisco
SEE MORE
INTERESTS
Beach
Romance
Last Minute
Family
SEE MORE
More featured deals
View all deals
From*
Student? You can travel cheaper!
Save
NCL - UP to $500 Free Onboard Spending
$199
Done


Google Maps - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://maps.google.com/
Google
Jim Gray
Search
Bookmarks
PageRank
Check
AutoLink
AutoFill
Send to
Jim
Gray
Jim Gray (computer scientist) - Wi...
HIV protein breaks biological clock...
Google Maps
Web Images Maps News Shopping Gmail more
Sign in
Google Maps
Search Maps
Show search options
Find businesses, addresses and places of interest. Learn more.
Get Directions
My Maps
Print
Send
New! Add driving directions to your website.
Set default location
Browse popular maps
Follow the Sundance Film Festival
See more maps
Put your business on Google Maps
Display your ads on Google Maps
Traffic
More...
Map
Satellite
Washington
Montana
North Dakota
Minnesota
South Dakota
Wisconsin
Michigan
Idaho
Wyoming
Nebraska
Iowa
Nevada
Utah
Colorado
Kansas
Missouri
Illinois
Indiana
Ohio
Pennsylvania
New York
California
Arizona
New Mexico
Oklahoma
Arkansas
Tennessee
Kentucky
West Virginia
Virginia
North Carolina
South Carolina
Georgia
Alabama
Mississippi
Louisiana
Texas
Florida
Gulf of Mexico
México
District of Columbia
Ottawa
Montréal
Toronto
Minneapolis
Milwaukee
Chicago
Detroit
Toledo
Cleveland
Buffalo
Philadelphia
Omaha
Lincoln
Kansas City
St Louis
Cincinnati
Denver
Salt Lake City
Sacramento
Stockton
Las Vegas
Los Angeles
Riverside
San Diego
Phoenix
Tucson
Albuquerque
El Paso
Wichita
Tulsa
Dallas
Austin
San Antonio
Houston
Atlanta
Jacksonville
Orlando
Miami
La Habana
Monterrey
Torreón
Aguascalientes
San Luis
500 mi
500 km
©2009 Google - Map data ©2009 Tele Atlas, LeadDog Consulting - Terms of Use
Find: interg
Next
Previous
Highlight all
Match case
Done


Edit View History Bookmarks Tools Help
http://www.ncbi.nlm.nih.gov/gquery/gquery.fcgi?itool=toolbar
Google
Search
Bookmarks
PageRank
Check
AutoLink
AutoFill
Send to
Genome
Entrez cross-database search
HIV protein breaks biological clock...
NCBI
Entrez, The Life Sciences Search Engine
HOME
SEARCH
SITE MAP
PubMed
All Databases
Human Genome
GenBank
Map Viewer
BLAST
Search across databases
GO
Clear
Help
Welcome to the Entrez cross-database search page
PubMed: biomedical literature citations and abstracts
PubMed Central: free, full text journal articles
Site Search: NCBI web and FTP sites
Books: online books
OMIM: online Mendelian Inheritance in Man
OMIA: online Mendelian Inheritance in Animals
Nucleotide: Core subset of nucleotide sequence records
EST: Expressed Sequence Tag records
GSS: Genome Survey Sequence records
Protein: sequence database
Genome: whole genome sequences
Structure: three-dimensional macromolecular structures
Taxonomy: organisms in GenBank
dbGaP: genotype and phenotype
UniGene: gene-oriented clusters of transcript sequences
CDD: conserved protein domain database
3D Domains: domains from Entrez Structure
UniSTS: markers and mapping data
PopSet: population study data sets
GEO Profiles: expression and molecular abundance

Motivating Example

Data: in Computer system, Data is any sort of information which is stored in computer memory.

Student ID	Student name	Course	Course Name	Grade
50000000	Peter Wong	CS3402	Database Systems	B+
50000000	Peter Wong	CS2302	Data Structures	A
50000001	Mary Tsui	CS3402	Database Systems	A-
50000002	Bob Lee	CS3402	Database Systems	B

- This way of storing data good?
- How to improve?

Introduction to DB Systems

- Motivations
  - File-processing Systems
    - permanent records stored in various files
    - application programs written to extract & add records
  - Disadvantages of traditional file-processing systems
    - data redundancy & inconsistency
    - difficulty in accessing data
    - data isolation & different data formats
    - concurrent access anomalies
    - security problem
    - integrity problem

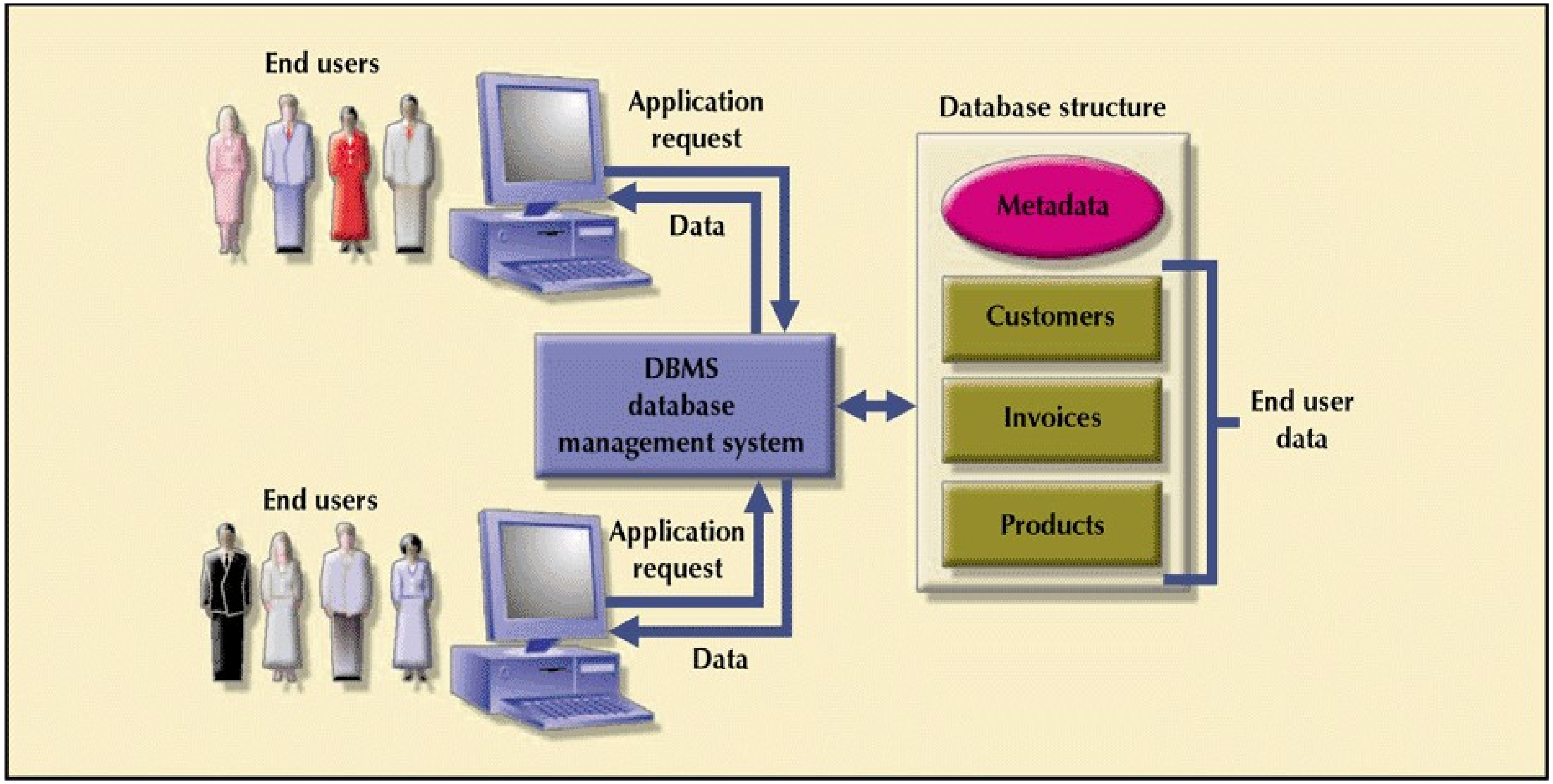
Introduction to DB Systems

- What is a Database (DB)?
  - A non-redundant, persistent collection of logically related records/files that are structured to support various processing and retrieval needs

- Database Management System (DBMS)
  - A set of software programs for creating, storing, updating, and accessing the data of a DB.

- Database System
  - an integrated system of hardware, software, people, procedures, and data
  - that define and regulate the collection, storage, management, and use of data within a database environment

Database Management System



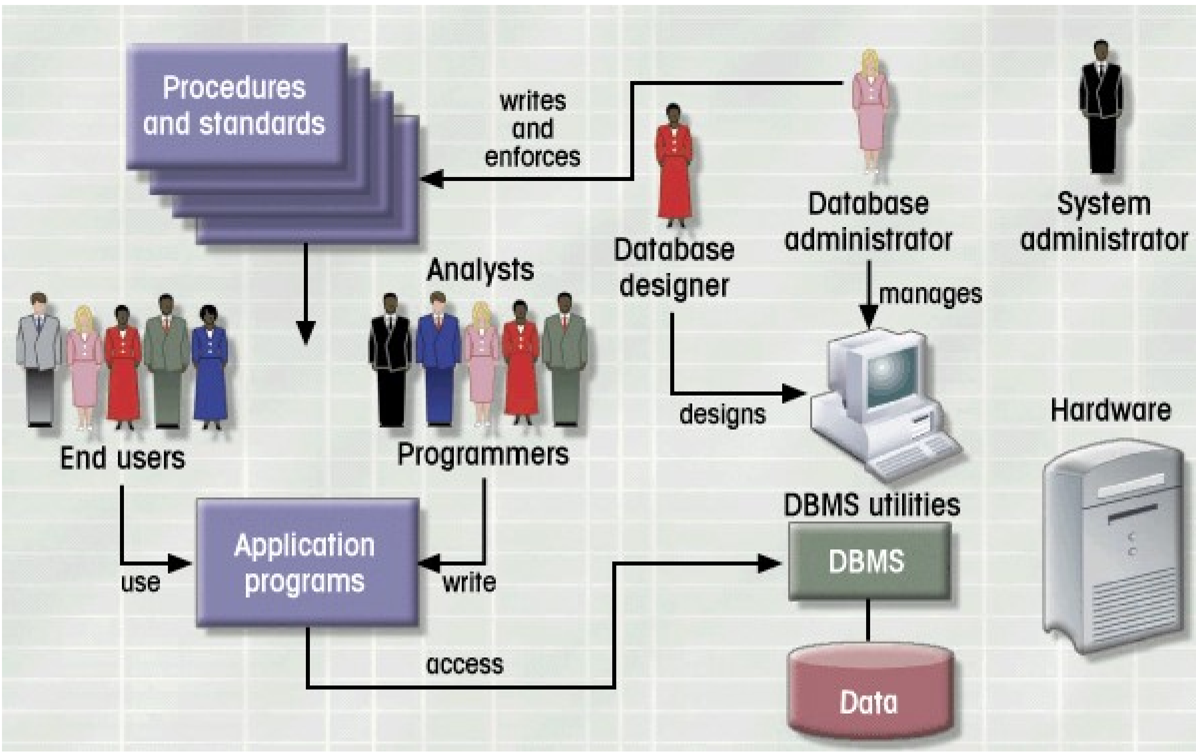
Database Systems: Design, Implementation, & Management: Rob & Coronel

- manages interaction between end users and database

Database Management System

- Difference between DBMS & other programming systems
  - the ability to manage persistent data
  - primary goal of DBMS: to provide an environment that is convenient, efficient, and robust to use in retrieving & storing data
- Other DBMS capabilities
  - data modeling
  - high-level languages to define, access and manipulate data
  - transaction management & concurrency control
  - access control
  - recovery

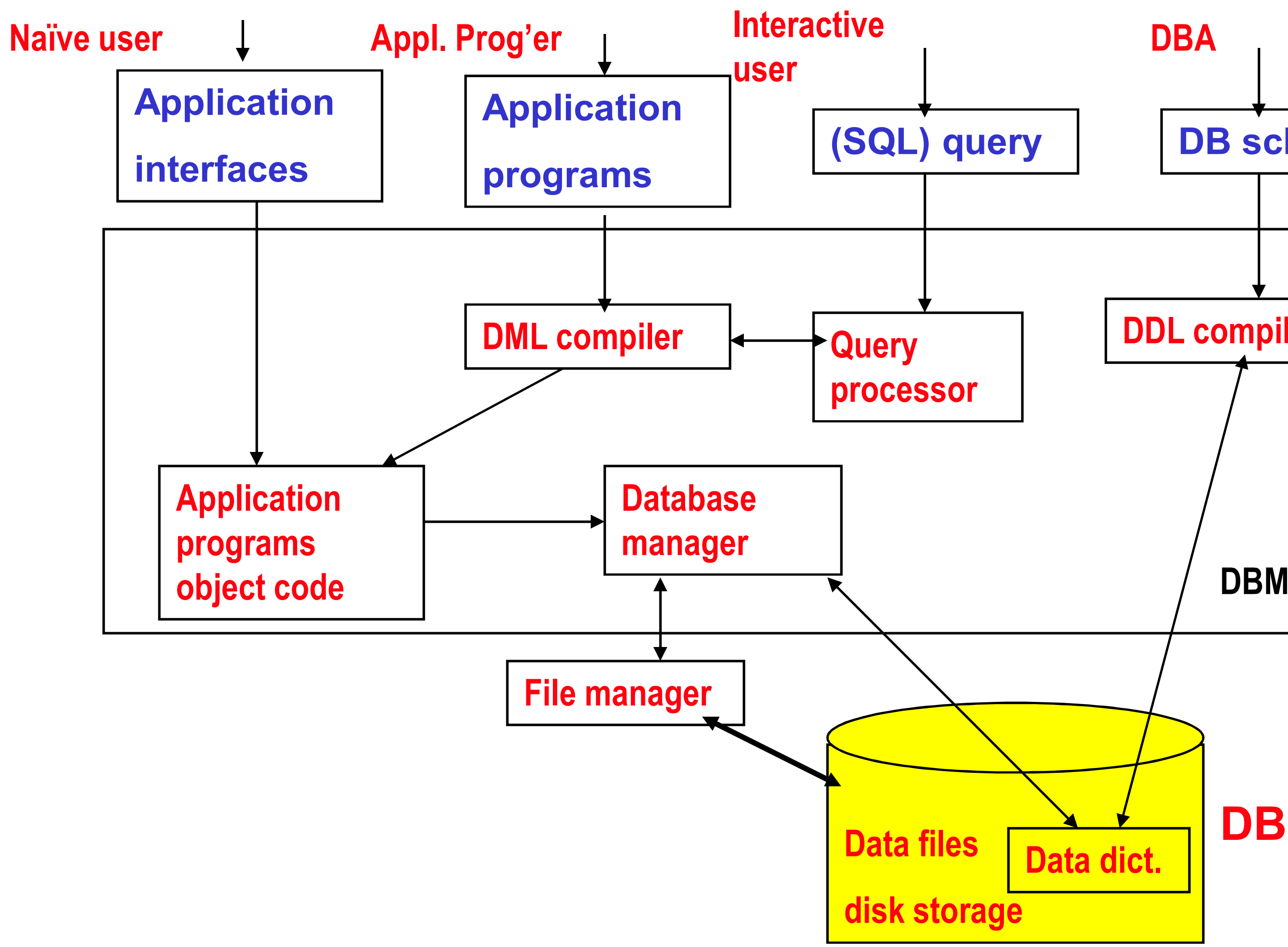
Database System



Database Systems: Design, Implementation, & Management: Rob & Coronel

- **Hardware**
- **Software**
  - **OS**
  - **DBMS**
  - **Applications**
- **People**
- **Procedures**
- **Data**

DB System Architecture



DB System Architecture

- *Data Definition Language (DDL)*

  - a language for defining DB schema

  - DDL statements compile to a data dictionary which is a file containing **metadata** (data about data), e.g., descriptions about the tables

- *Data Manipulation Language (DML)*

  - a language that enables users to manipulate data

  - an important subset for retrieving data is called Query Language

  - two types of DML: ***procedural*** (specify "what" & "how") vs. ***declarative*** (just specify "what")

DB System Architecture

- *Database Administrator (DBA)*
  - DBA is the person who has central control over the DB
  - Main functions of DBA:
    - schema definition
    - storage structure and access method definition
    - schema and physical organization modification
    - granting of authorization for data access
    - integrity constraint specification

DB System Architecture

- *Database Users*
  - Application Programmers
    - Writing embedded DML in a host language
  - Interactive Users:
    - Using query languages
  - Naive Users:
    - Running application programs

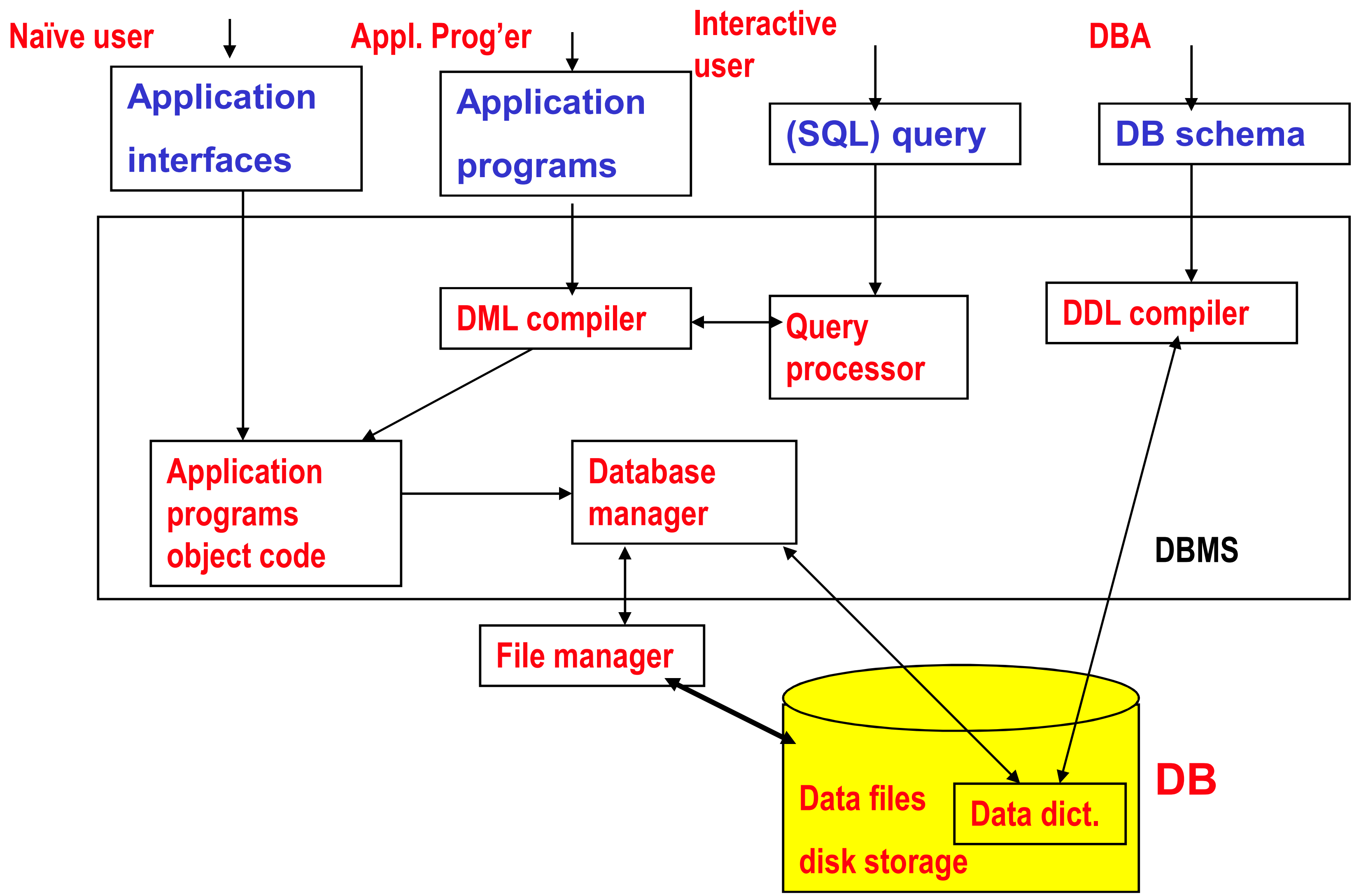
DB System Architecture

- File Manager
  - allocation of space
  - operations on files
- DB Manager
  - interface between stored data and application programs/queries
  - translate conceptual level commands into physical level ones
  - responsible for
    - access control
    - concurrency control
    - backup & recovery
    - integrity

DB System Architecture

- Query Processor
  - translate high-level queries into low-level instructions
  - query optimization
- DML (Pre)compiler
  - translates DML statements embedded in application program into procedure calls
- DDL (Pre)compiler
  - converts DDL statements to data dictionary items (eg, table descriptions)

DB System Architecture

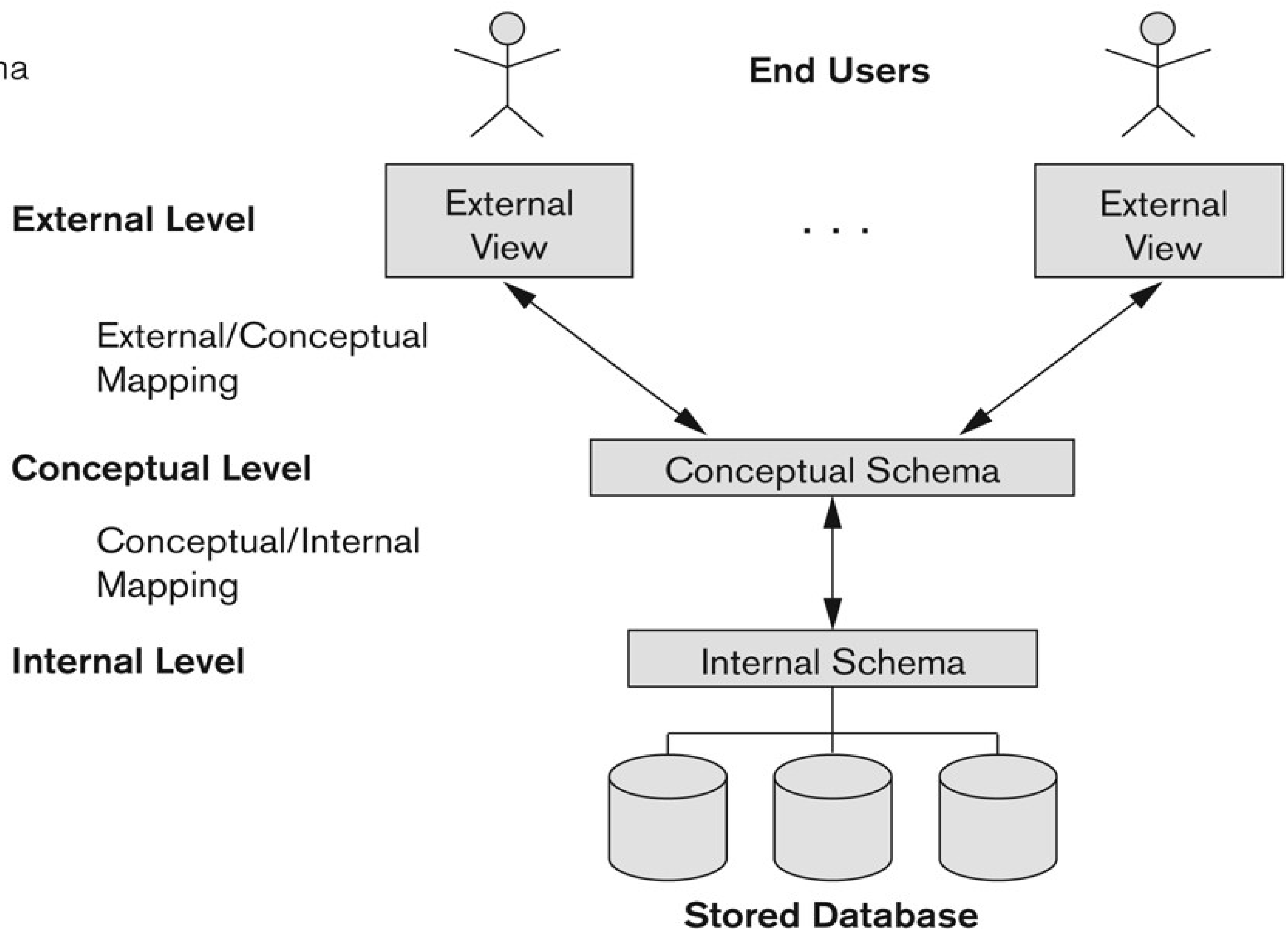


Data Abstraction

- Data Abstraction
  - Abstract view of the data
    - simplify interaction with the system
    - hide details of how data is stored and manipulated
  - Levels of abstraction
    - view/external level: partial schema
    - conceptual level: schema, what data are actually stored
    - physical/internal level: data structures; how data are actually stored

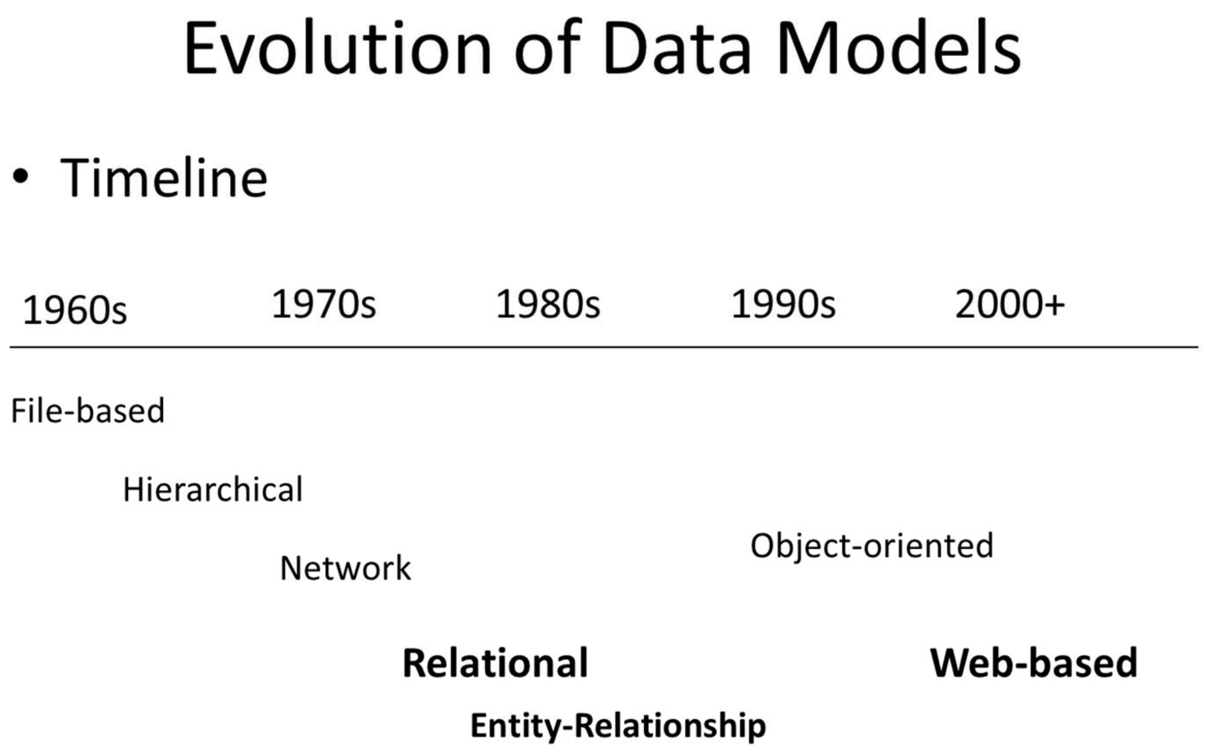
Data Abstraction: 3-level architecture

Figure 2.2
The three-schema architecture.



Data Models

- Data Model (conceptual level)
  - A collection of conceptual tools for describing data, data relationships, operations, and consistency constraints
  - the "core" of a database



Data Independence

- the ability to modify a schema definition in one level without affecting a schema in the next higher level
- there are two kinds (a result of the 3-level architecture):
  - ***physical data independence***

    -- *the ability to modify the physical schema without altering the conceptual schema and thus, without causing the application programs to be rewritten*
  - ***logical data independence***

    -- *the ability to modify the conceptual schema without causing the application programs to be rewritten*

Course Objectives

- ER model: characterize relationships among entities
- Relational model: transform from ER diagram to tables
- SQL: language for writing queries
- Relational Algebra: logical way to represent queries
- Normal Forms: how to design good tables
- File Organization: provide file level structure to speed up query
- Query Optimization: transform queries into more efficient ones
- Transactions and Concurrency Control: handle concurrent operations and guarantee correctness of the database

Course Objectives

- ER model: characterize relationships among entities
- Relational model: transform from ER diagram to tables
- SQL: language for writing queries
- Relational Algebra: logical way to represent queries
- Normal Forms: how to design good tables
- File Organization: provide file level structure to speed up query
- Query Optimization: transform queries into more efficient ones
- Transactions and Concurrency Control: handle concurrent operations and guarantee correctness of the database

The Entity-Relationship Model

- Preliminaries
  - Proposed by P. Chen in 1976
  - Direct, easy-to-understand graphical notation
  - Translates readily to relational schema for database design

Ideas/requirements → E/R design → Relational schema → Relational database

- **Three basic concepts:**

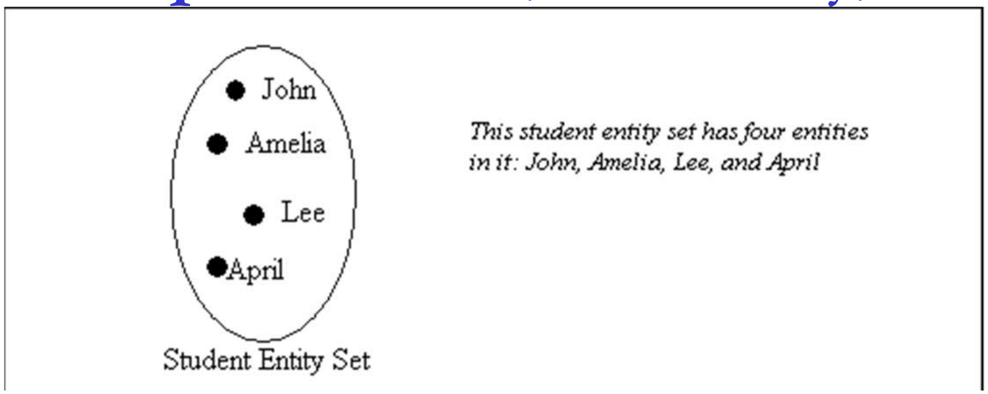
  Entity, Attribute, Relationship

ER Model Concepts

- **Entity**
  - a distinguishable object with an independent existence

    Example: John Chan, CityU, HSBC, …
- **Entity Set**
  - a set of entities of the same type

    Example: Student, University, Bank, ...



ER Model Concepts

- **Attribute**(Property) -- a piece of information describing an entity
  - *Example*: Name, ID, Address, DoB are attributes of a student entity
  - Each attribute can take a value from a domain

    Example: Name $\in$ Character String,

    ID $\in$ Integer, ...
  - Formally, an attribute **A** is a function which maps from an entity set **E** into a domain **D**:

$$\mathbf{A: E \rightarrow D}$$

Types of Attributes

- Simple
  - Each entity has a single atomic value for the attribute. For example, SSN or Sex, name…
- Composite
  - The attribute may be composed of several components. For example:
    - Address(Flat, Block, Street, City, State, Country)
    - Composition may form a hierarchy where some components are themselves composite
- Multi-valued
  - An entity may have multiple values for that attribute. For example, Color of a CAR or PreviousDegrees of a STUDENT
    - Denoted as {Color} or {PreviousDegrees}
    - E.g., {{BSc, 1990}, {MSc, 1993}, {PhD, 1998}}

Example of a composite attribute

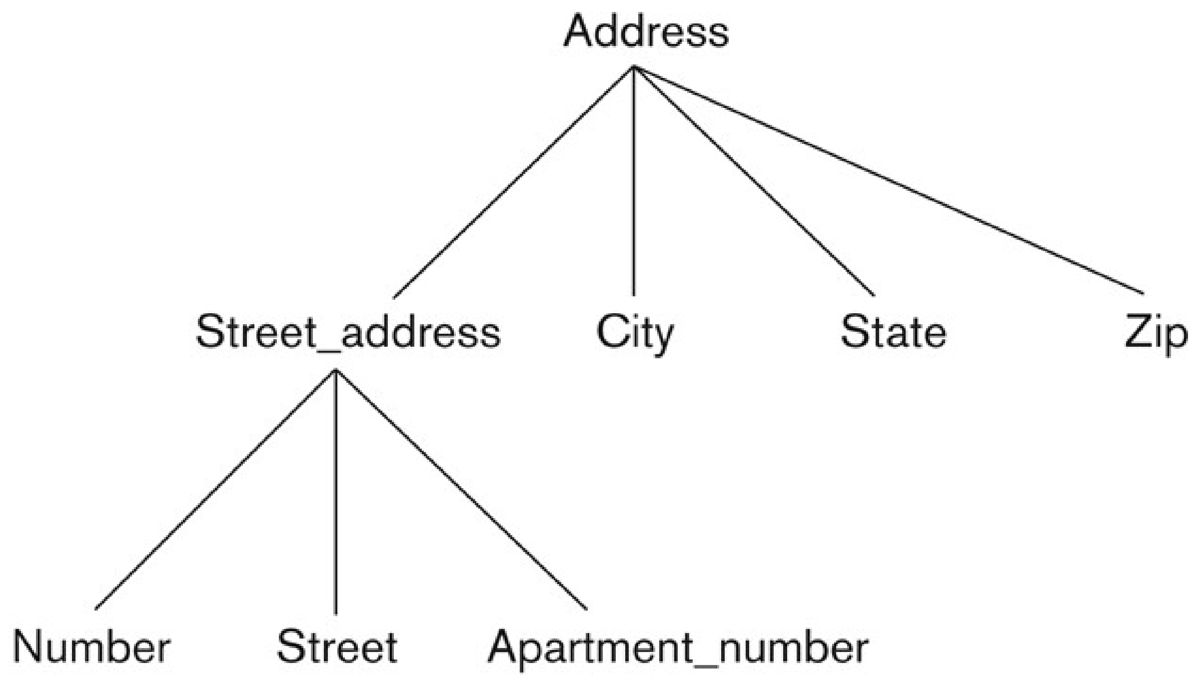


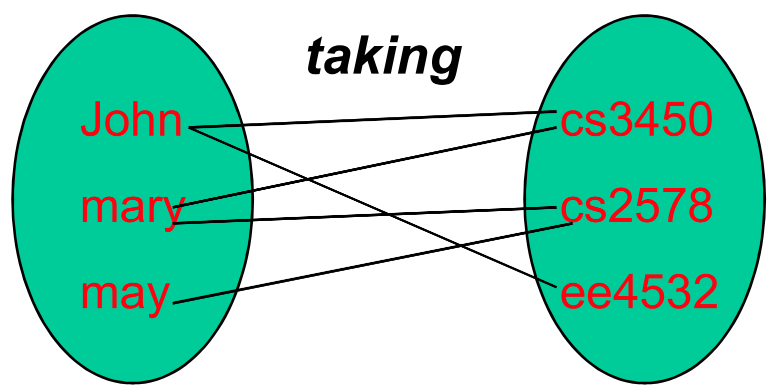
Figure 3.4
A hierarchy of composite attributes.

Types of Attributes

- In general, composite and multi-valued attributes may be nested to any number of levels
  - For example, PreviousDegrees of a STUDENT is a composite multi-valued attribute denoted by {PreviousDegrees (College, Year, Degree, Field)}
  - Multiple PreviousDegrees values can exist
  - Each has four subcomponent attributes:
    - College, Year, Degree, Field

ER Model Concepts

- **Relationship** -- an association among several entities
  - *Example*: Patrick and Eva are friends
    Patrick is taking cs3450
- **Relationship Set** -- a set of relationships of the same type
  - Example:



  - Formally, a relationship **R** is a subset of:
    $\{ (\mathbf{e1}, \mathbf{e2}, \ldots, \mathbf{ek}) \mid \mathbf{e1} \in E1, \mathbf{e2} \in E2, \ldots, \mathbf{ek} \in Ek) \}$

ER Model Concepts

- **Relationship** vs. **Attribute**
  - an attribute **A: E** $\rightarrow$ **D** is a "simplified" form of a relationship:
    - If we allow **D** to be an Entity Set, then **A** becomes a relationship
  - a relationship can carry attributes
    - properties of the relationship
    - <u>Example</u>: Patrick takes cs2450 with a grade of B+
      Supplier S supplies item T with a price of P

ER Model Concepts

- **Entity Set** vs. **Attribute**
  - When to use attribute, and when to use entity set?
    - <u>Example</u>: Employee and Phone

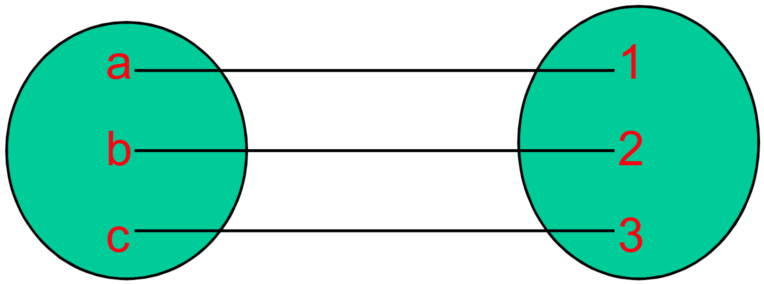
      1) *employee* entity set with attribute *phone#*

      2) *empPhn* relationship set with entity sets *employee* and *phone#*
  - No simple answer, depending on
    - what we want to model
    - meaning of attributes

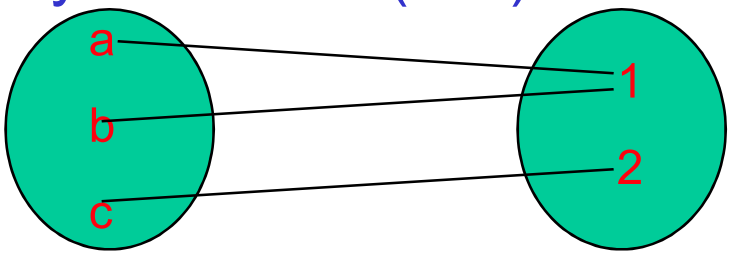
ER Model Concepts

- Integrity Constraints: are the requirements (i.e., rules) imposed by a relationship that must be met when updating the database.

    - Mapping Cardinalities
        - One - to - One (1:1)



        - One - to - Many (1:M) / Many - to - One (N:1)



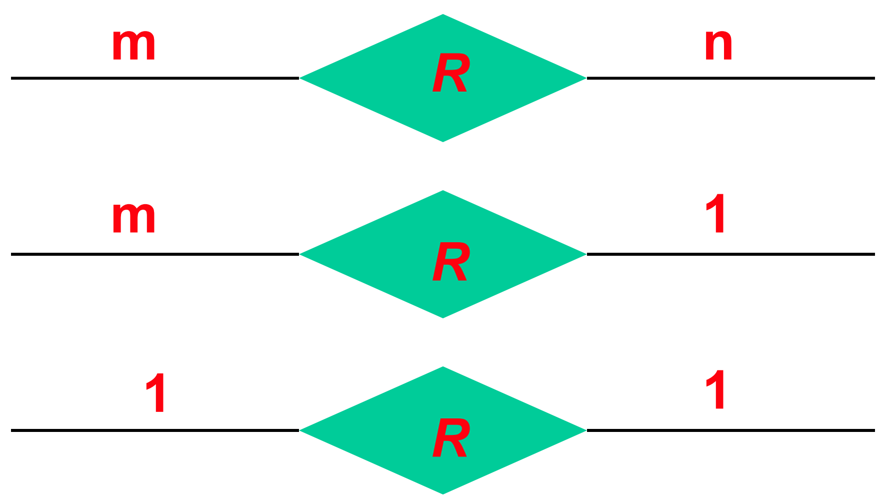
        - Many - to - Many (M:N) ?

ER Model Concepts

- Integrity Constraints (cont'd)
  - Keys: to distinguish individual entities or relationships
    - **superkey** -- a set of one or more attributes which, taken together, identify uniquely an entity in an entity set
      - Example: {student ID, Name} identify a student
    - **candidate key** -- *minimal* set of attributes which can identify uniquely an entity in an entity set
      - a superkey for which no proper subset is a superkey
      - Example: student ID identify a student,
but Name is not a candidate key (WHY?)
    - **primary key** -- a candidate key chosen by the DB designer to identify an entity in an entity set

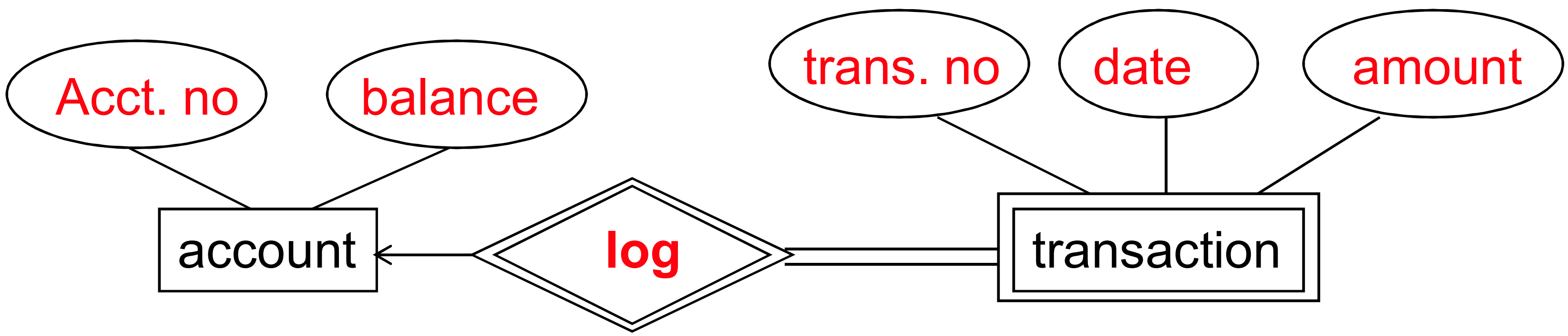
ER Model Diagram

- ER Diagram
  - Rectangles: Entity Sets
  - Ellipses: Attributes
  - Diamonds: Relationship Sets
  - Lines: Attributes to Entity/Relationship Sets or, Entity Sets to Relationship Sets

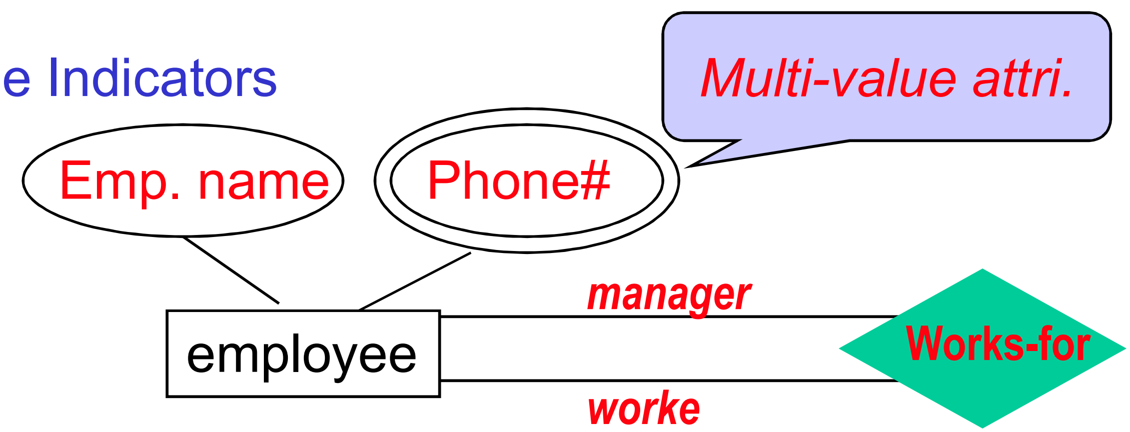


ER Model Diagram

- Weak Entity Set
  - an entity set that does NOT have enough attributes to form a primary/candidate key

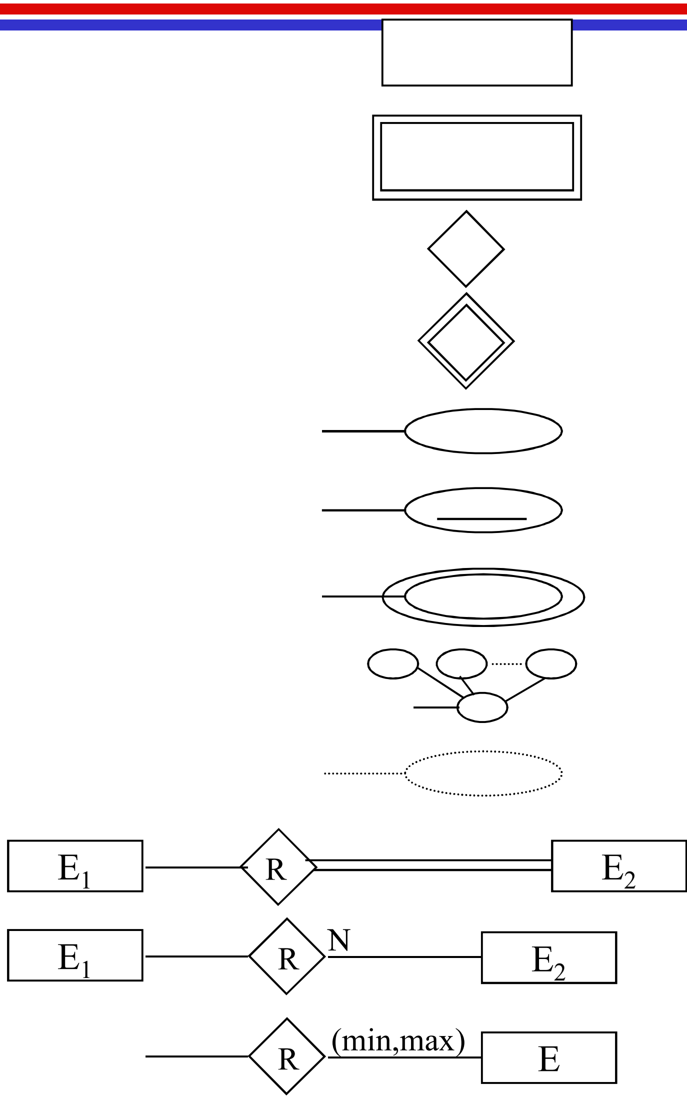


- Role Indicators



Summary of ER-Diagram Notation

Symbol



Meaning

ENTITY TYPE

WEAK ENTITY TYPE

RELATIONSHIP TYPE

IDENTIFYING RELATIONSHIP TYPE

ATTRIBUTE

KEY ATTRIBUTE

MULTIVALUED ATTRIBUTE

COMPOSITE ATTRIBUTE

DERIVED ATTRIBUTE

TOTAL PARTICIPATION OF E_2 IN R

CARDINALITY RATIO 1:N FOR E_1:E_2 IN R

STRUCTURAL CONSTRAINT (min, max) ON PARTICIPATION OF E IN R

Example COMPANY Database

- We need to create a database schema (definition) based on the following (simplified) application requirements of the COMPANY Database
- Refer to textbook for more detailed description of requirements:
    - The company is organized into DEPARTMENTs
    - Each DEPARTMENT has a name, number and an EMPLOYEE who *manages* the department
    - We keep track of the start date of the department manager. A department may have several locations
    - Each DEPARTMENT controls/has a number of PROJECTs
    - Each project has a unique name, unique number and is located at a single location

Example COMPANY Database

- The database will store each EMPLOYEE's social security number (ssn), address, salary, sex, and birthdate
  - Each employee *works for* one department but may *work on* several projects
  - The DB will keep track of the number of hours per week that an employee currently works on each project
  - It is required to keep track of the *direct supervisor* of each employee

- Each employee may *have* a number of DEPENDENTs
  - For each dependent, the DB keeps a record of name, sex, birthdate, and relationship to the employee

Initial Conceptual Design of Entity Sets for the COMPANY Database Schema

- Based on the requirements, we can identify four initial entity sets in the COMPANY database:
  - DEPARTMENT
  - PROJECT
  - EMPLOYEE
  - DEPENDENT
- The initial attributes shown are derived from the requirements description

Initial Design of Entity Sets:
EMPLOYEE, DEPARTMENT, PROJECT, DEPENDENT

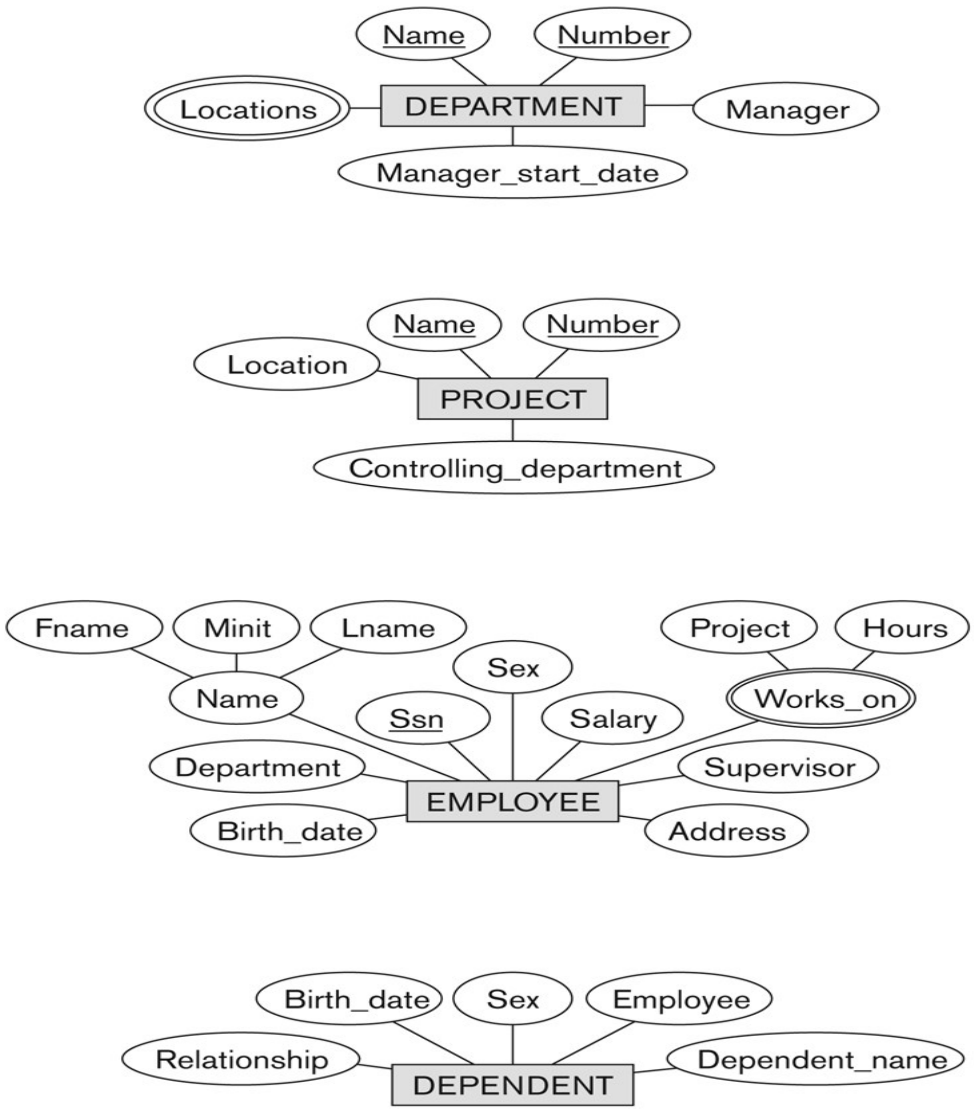
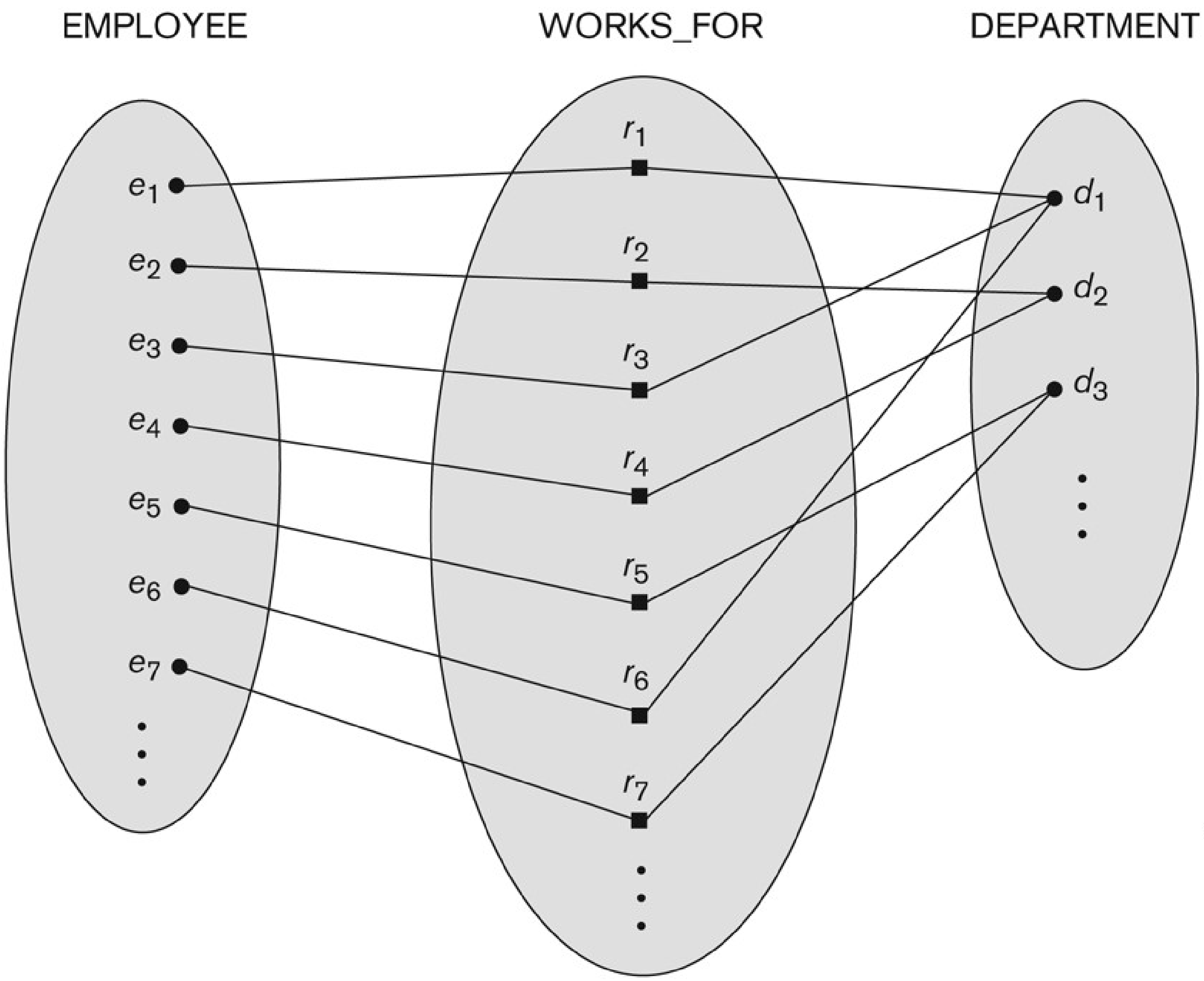


Figure 3.8
Preliminary design of entity types for the COMPANY database. Some of the shown attributes will be refined into relationships.

Relationship instances of the WORKS_FOR N:1 relationship between EMPLOYEE and DEPARTMENT

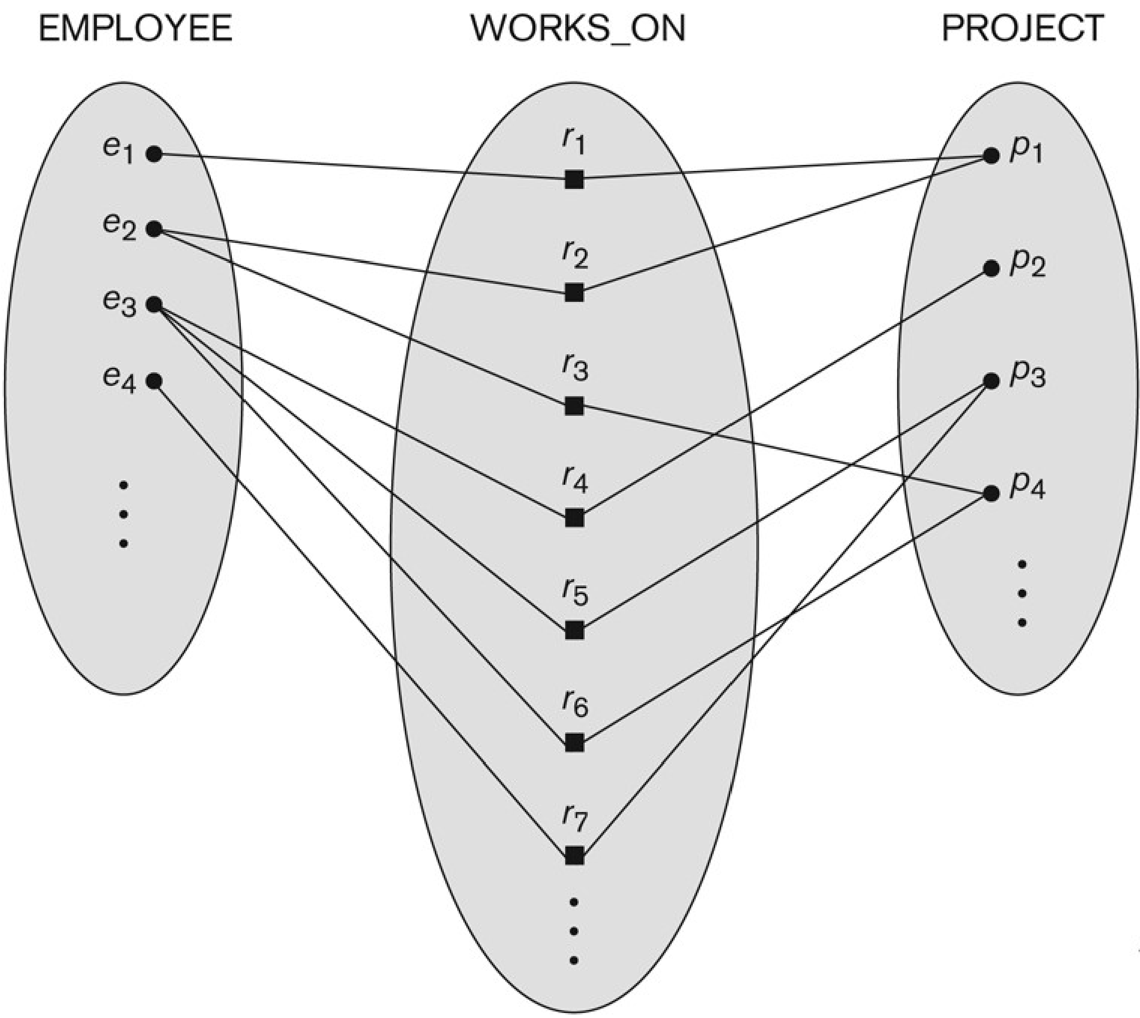


N-to-1

Many employees in the same department

Figure 3.9
Some instances in the WORKS_FOR relationship set, which represents a relationship type WORKS_FOR between EMPLOYEE and DEPARTMENT.

Relationship instances of the M:N WORKS_ON relationship between EMPLOYEE and PROJECT

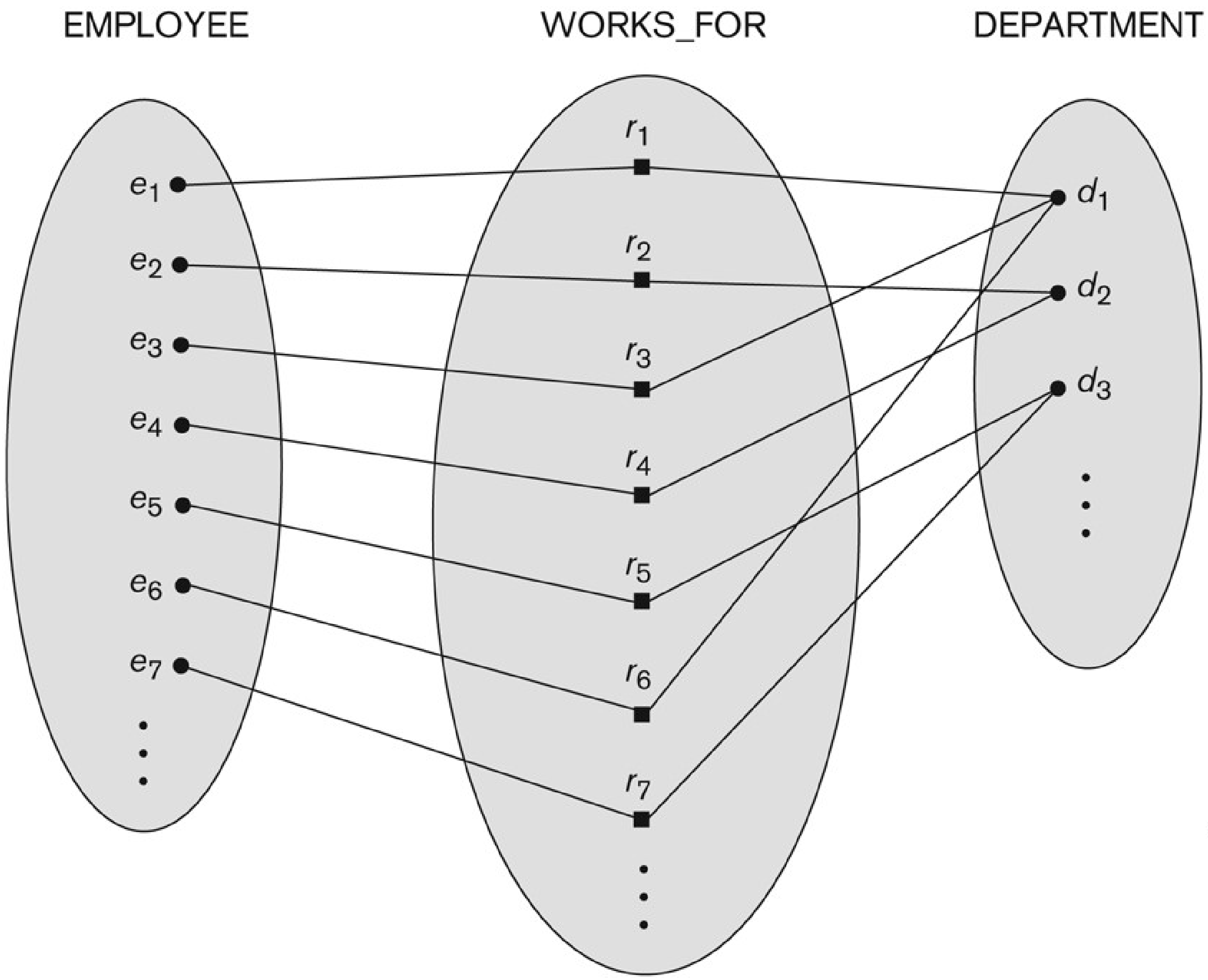


M-to-n

Many employee work on many projects

Figure 3.13
An M:N relationship, WORKS_ON.

Many-to-one (N:1) Relationship



Select anyone in EMPLOYEE, one can be identified in DEPARTMENT

One-to-one is special case of Many-to-one

Figure 3.9
Some instances in the WORKS_FOR relationship set, which represents a relationship type WORKS_FOR between EMPLOYEE and DEPARTMENT.

Many-to-many (M:N) Relationship

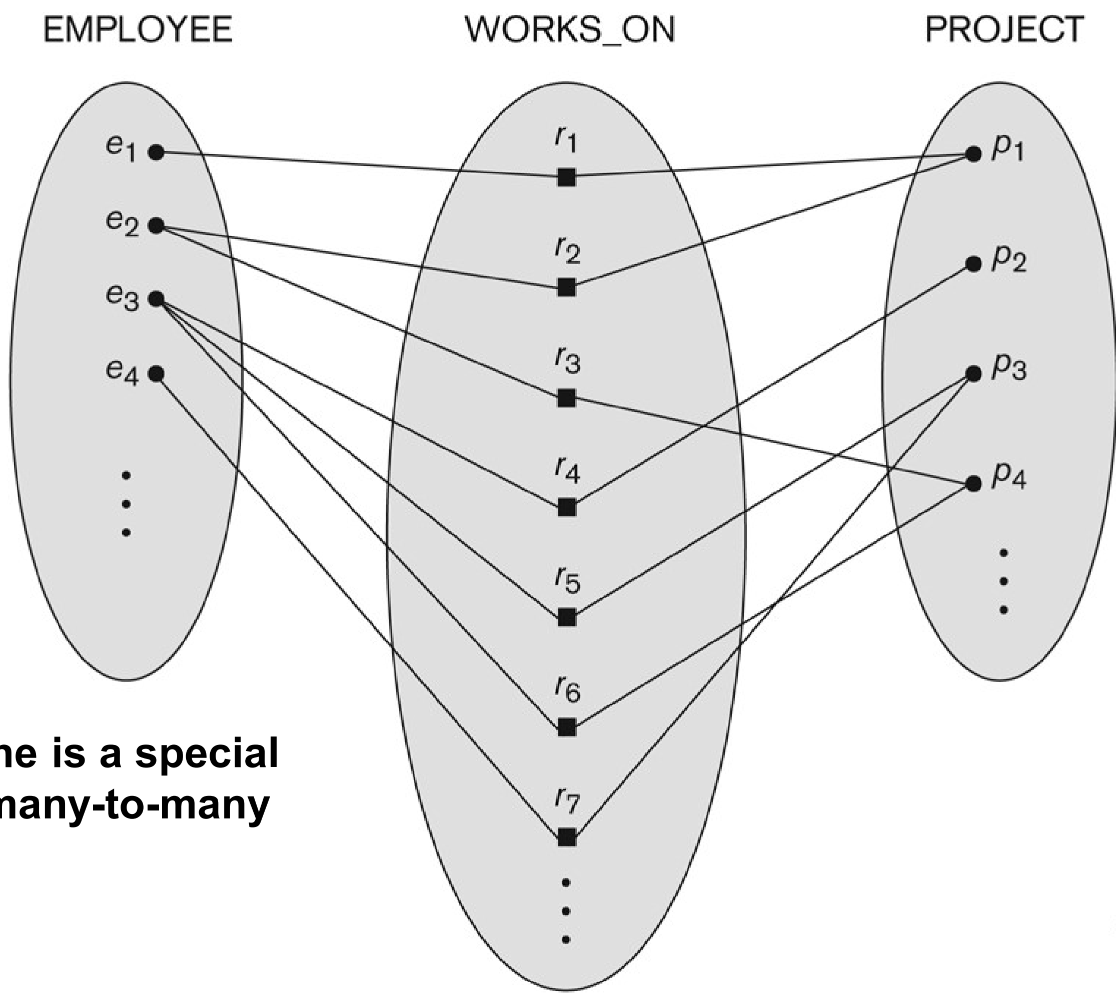
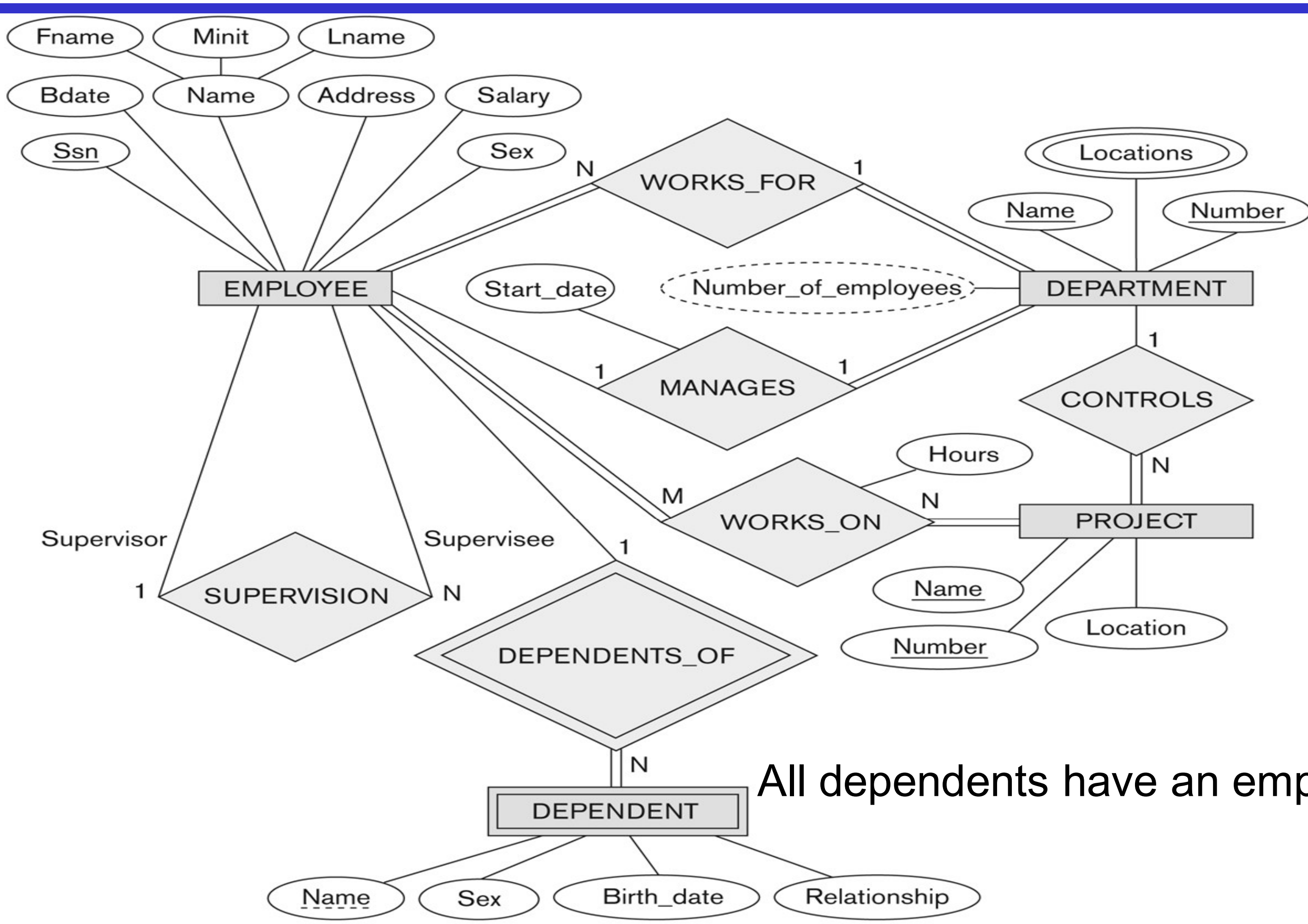


Figure 3.13
An M:N relationship, WORKS_ON.

Many-to-one is a special case of a many-to-many

ER DIAGRAM – Relationship Types are:

WORKS_FOR, MANAGES, WORKS_ON, CONTROLS, SUPERVISION, DEPENDENTS_OF



All dependents have an employee

Figure 3.2
An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter.

Summary

- The ER Model is regarded as the 1st "conceptual/semantic" model centered around relationships, not attributes
- It combines successfully the best features of the previous data models
- simple and easy to understand
- can be mapped to tables (relational model) in a straightforward manner

  (to be studied in the coming lecture series)