CMU SCS

15-826: Multimedia Databases and Data Mining

Lecture#1: Introduction

Christos Faloutsos
CMU
www.cs.cmu.edu/~christos

CMU SCS

Outline

Goal: 'Find similar / interesting things'

- Intro to DB
- Indexing - similarity search
- Data Mining

CMU SCS

Problem

Given a large collection of (multimedia) records, find similar/interesting things, ie:

- Allow fast, approximate queries, and
- Find rules/patterns

CMU SCS

Sample queries

- Similarity search
  - Find pairs of branches with similar sales patterns
  - find medical cases similar to Smith's
  - Find pairs of sensor series that move in sync
  - Find shapes like a spark-plug
  - (nn: 'case based reasoning')

CMU SCS

Sample queries –cont'd

- Rule discovery
  - Clusters (of branches; of sensor data; ...)
  - Forecasting (total sales for next year?)
  - Outliers (eg., unexpected part failures; fraud detection)

CMU SCS

Outline

Goal: 'Find similar / interesting things'

- Intro to DB
- Indexing - similarity search
- Data Mining

CMU SCS

Detailed Outline

Intro to DB

- Relational DBMS - what and why?
  - inserting, retrieving and summarizing data
  - views; security/privacy
  - (concurrency control and recovery)
- Object-Relational DBMS - what and why?

CMU SCS

What is the goal of rel. DBMSs

CMU SCS

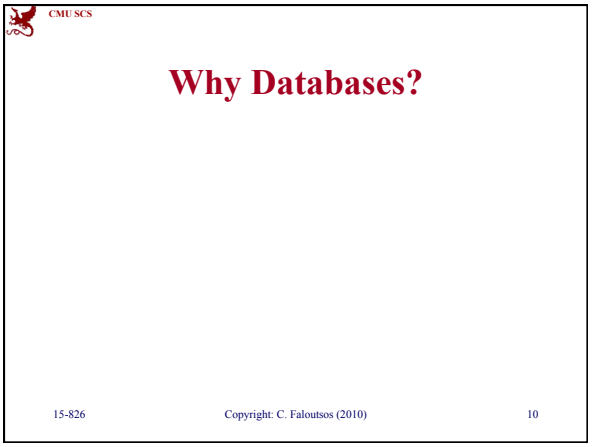
What is the goal of rel. DBMSs

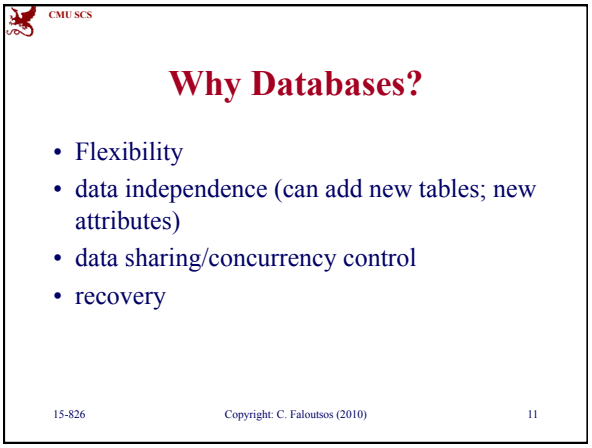
Electronic record-keeping:

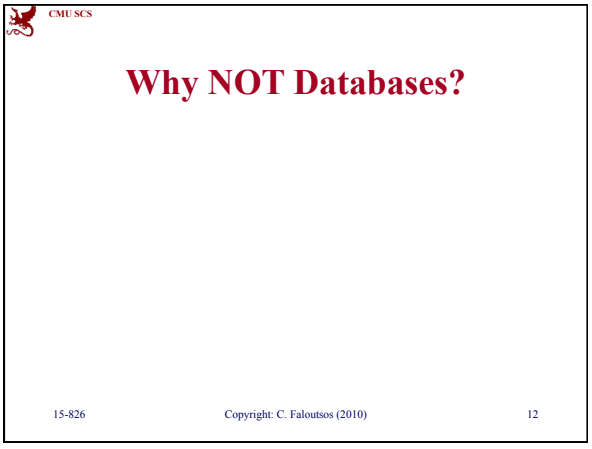
Fast and convenient access to information.

Eg.: students, taking classes, obtaining grades;

- find my gpa
- <and other ad-hoc queries>


CMU SCS
Why Databases?
15-826
Copyright: C. Faloutsos (2010)
10


CMU SCS
Why Databases?
• Flexibility
• data independence (can add new tables; new attributes)
• data sharing/concurrency control
• recovery
15-826
Copyright: C. Faloutsos (2010)
11


CMU SCS
Why NOT Databases?
15-826
Copyright: C. Faloutsos (2010)
12

CMU SCS

Why NOT Databases?

- Price
- additional expertise (SQL/DBA)
- over-kill for small data sets

CMU SCS

Main vendors/products

Commercial
- Oracle
- IBM/DB2
- MS SQL-server
- Sybase
- (MS Access,
- ...)

Open source
Postgres (UCB)
mySQL, sqlite, mSQL
miniBase (Wisc)
Predator (Cornell)
(www.sigmod.org)

CMU SCS

Detailed Outline

Intro to DB
- Relational DBMS - what and why?
  - inserting, retrieving and summarizing data
  - views; security/privacy
  - (concurrency control and recovery)
- Object-Relational DBMS - what and why?

CMU SCS

How do DBs work?

We use **`sqlite3`** as an example, from

`http://www.sqlite.org`

CMU SCS

How do DBs work?

%sqlite3 mydb # mydb: file

sql>create table student (

ssn fixed;

name char(20));

student	
ssn	name

CMU SCS

How do DBs work?

sql>insert into student
values (123, “Smith”);

sql>select * from student;

student	
ssn	name
123	Smith

CMU SCS

How do DBs work?

sql>create table takes (
ssn fixed,
c_id char(5),
grade fixed));

takes		
ssn	c_id	grade

CMU SCS

How do DBs work - cont'd

More than one tables - joins
Eg., roster (names only) for 15-826

student	
ssn	name

takes		
ssn	c_id	grade

CMU SCS

How do DBs work - cont'd

sql> select name
from student, takes
where student.ssn = takes.ssn
and takes.c_id = "15826"

CMU SCS

SQL-DML

General form:

select a1, a2, … an
from r1, r2, … rm
where P
[**order by** ….]
[**group by** …]
[**having** …]

CMU SCS

Aggregation

Find ssn and GPA for each student

student	
ssn	name

takes		
ssn	c_id	grade
123	603	4
123	412	3
234	603	3

CMU SCS

Aggregation

sql> select ssn, grade
from takes;

takes		
ssn		grade
123		4
123		3
234		3

CMU SCS

Aggregation

sql> select ssn, avg(grade)
from takes;

WRONG

CMU SCS

Aggregation

sql> select ssn, avg(grade)
from takes
group by ssn;

takes

ssn	c_id	grade
123	603	4
123	412	3
234	603	3

ssn	avg(grade)
123	3.5
234	3

CMU SCS

Detailed Outline

Intro to DB

- Relational DBMS - what and why?
  - inserting, retrieving and summarizing data
  - views; security/privacy
  - (concurrency control and recovery)
- Object-Relational DBMS - what and why?

CMU SCS

Views - what and why?

- suppose you ONLY want to see ssn and GPA (eg., in your data-warehouse)
- suppose secy is only allowed to see GPAs, but not individual grades
- -> VIEWS!

CMU SCS

Views

sql> create view fellowship as (
select ssn, avg(grade)
from takes group by ssn);

takes

ssn	c_id	grade
123	603	4
123	412	3
234	603	3

ssn	avg(grade)
123	3.5
234	3

CMU SCS

Views

sql> create view fellowship as (
select ssn, avg(grade)
from takes group by ssn);

takes

ssn	c_id	grade
123	603	4
123	412	3
234	603	3

ssn	avg(grade)
123	3.5
234	3

CMU SCS

Views

Views = 'virtual tables'

CMU SCS

Views

sql> select * from fellowship;

takes		
ssn	c_id	grade
123	603	4
123	412	3
234	603	3

ssn	avg(grade)
123	3.5
234	3

CMU SCS

Views

sql> grant select on fellowship to secy;

takes		
ssn	c_id	grade
123	603	4
123	412	3
234	603	3

ssn	avg(grade)
123	3.5
234	3

CMU SCS

Detailed Outline

Intro to DB

- Relational DBMS - what and why?
  - inserting, retrieving and summarizing data
  - views; security/privacy
  - (concurrency control and recovery)
- Object-Relational DBMS - what and why?

CMU SCS

Detailed Outline

Intro to DB

- Relational DBMS - what and why?
  - inserting, retrieving and summarizing data
  - views; security/privacy
  - (concurrency control and recovery)
- Object-Relational DBMS - what and why?

CMU SCS

Why more than RDBMSs?

- RDBMS: tuples, of numbers + strings
- What apps need only those?

CMU SCS

Why more than RDBMSs?

- RDBMS: tuples, of numbers + strings
- What apps need only those?
  - Banks
  - Airlines
  - Retailer stores
  - ...
- Q: Other apps, with more req's?

CMU SCS

Why more than RDBMS's

- Q: Other apps, with more req's?
- A:
  - text
  - multimedia; financial apps/forecasting
  - Geographic Inf. Sys.
  - CAD/CAM
  - Network management

CMU SCS

Ideally, we'd like to:

- create a new data type (eg., 'image', 'time-sequence')
- define functions on it (like (dist(im1, im2))
- be able to ask queries like

```
select * from employee
where dist(employee.face, given-face) <= 10;
```

CMU SCS

OR DBMSs

traditional DBMS + attempts to provide
- user defined data types
- support for large / complex objects
- (inheritance - ISA hierarchies)

CMU SCS

SQL-3 extensions

- complex types (sets, lists, multisets)
- inheritance (IS-A hierarchies)
- User Defined Functions (UDFs)

CMU SCS

Complex types

Sample syntax

eg,

```
create type MyDate (
   day decimal(2),
   month char(3),
   year decimal (4)
);
```

CMU SCS

BLObs etc:

- Large objects, eg., video, images, 3d-MRI scans
- new data types: LOB (=Large OBject)
  - BLOB: (up to 4Gb; binary: jpeg, mpeg, ...)
  - CLOB: (up to 2Gb; character: english text)
  - NCLOB:(.....; multi-byte characters)

CMU SCS

Stored procedures

Sample syntax

```
SQL> create or replace procedure del-st-rec
       (s-id number) as
        begin
          delete from student
          where s-id = ssn;
        end del-st-rec;
SQL> execute del-st-rec ( 123 );
```


CMU SCS

Conclusions

- (relational) DBMSs: electronic record keepers
- customize them with **create table** commands
- ask SQL queries to retrieve info

CMU SCS

Conclusions cont'd

main advantages over flat files & scripts:

- logical + physical data independence (ie., flexibility of adding new attributes, new tables and indices)
- concurrency control and recovery for free

CMU SCS

Conclusions cont'd

- OR-DBMS: user-defined data types (eg., images), and U.D. functions.

CMU SCS

For more info:

- Microsoft Access: available on ANDREW clusters (PC)
- Sqlite3: **www.sqlite.org**
- postgres:
  http://www.postgresql.org/docs/
- Ramakrishna + Gehrke, 3rd edition
- 15-415 web page, eg,
  - http://www.cs.cmu.edu/~christos/courses/dbms-F09/