CSC7072: Databases, fall 2015

Dr. Kim Bauters

introduction and course outlines

introduction and course outlines

introduction

where are databases?

introduction

why databases?

introduction and course outlines

introduction

consider this problem:

A university needs to keep track of all its students and employees. Students may be enrolled in a degree program or in multiple courses. Employees may be researchers, and some employees may teach multiple courses. These are thought in various rooms spread over the campus. Employees who are researchers can be assigned grants. Employees are paid based on their post and their years of experience.

How to keep track of all this information in a manageable way?

introduction and course outlines

introduction

DataBase Management System (DBMS)

- contains interrelated data
  for example: customers, products we sell, orders by customers
- offers a set of programs to access the data
- offers an environment that is both *convenient* and *efficient*

DBMS are used *everywhere*:

universities

banking

airlines

government

stores

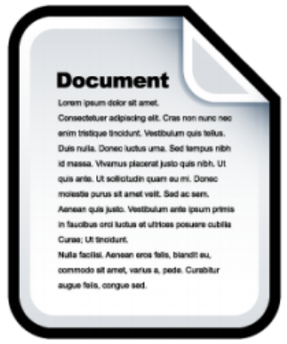
manufacturing

introduction and course outlines

introduction

but how do we create a DBMS?

early days: store as a files on a computer



had its share of problems

introduction and course outlines

introduction

but how do we create a DBMS?

early days: store as a files on a computer

had its share of problems: ***redundancy***

introduction and course outlines

introduction

but how do we create a DBMS?

early days: built on top of file system of computer

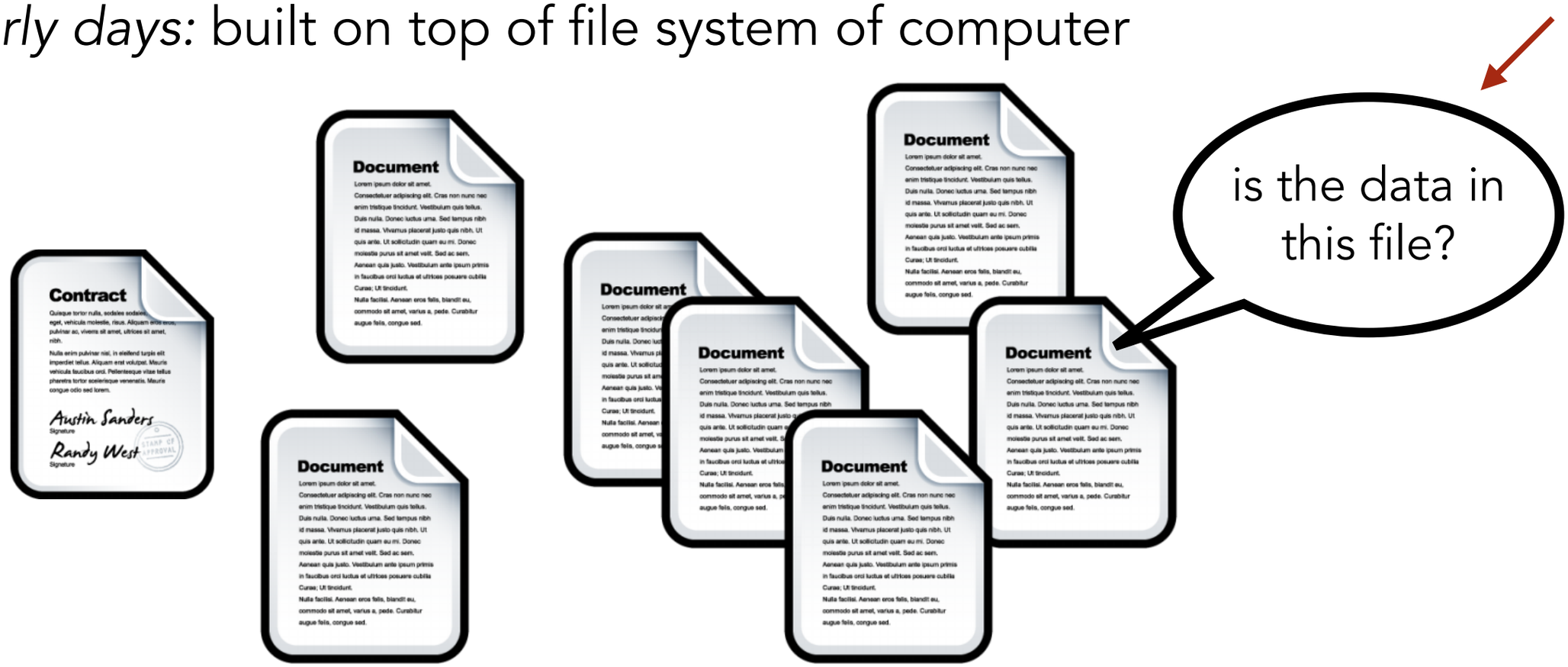
had its share of problems: *redundancy,* ***inconsistency***

introduction and course outlines

introduction

but how do we create a DBMS?

early days: built on top of file system of computer



had its share of problems: *redundancy, inconsistency,* ***data access***

introduction and course outlines

introduction

but how do we create a DBMS?

early days: built on top of file system of computer

and a lot of other problems you might not think about:

- everyone stored files in their own way
  so a new application needed for every company
- hard to impose constraints
  from now on, for clarity, all family names must be in UPPERCASE
- no protection against failures, lack of concurrent access
  power is turned off while you were modifying two files
- security is a problem
  how to prevent access? give access to only some of the data?

introduction and course outlines

introduction

what is a database?

technical: hardware and software to manage the storage and retrieval of (large sets of) interrelated data

business: place to safely/securely store and access valuable information

intuitive: a glorified collection of linked tables

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

introduction and course outlines

module outlines

- brief look at data models in modern DBMS *(part 02)*
- data retrieval / manipulation using SQL *(part 03, part 04)*
- database design: Entity-Relation (ER) model *(part 05, part 07)*
- data definition using SQL *(part 06)*
- database theory: normalisation *(part 08)*
- *database applications:* data mining *(part 09)*
- *database applications:* transactions *(part 10)*

introduction and course outlines

module outlines

lecturer:

Dr Kim Bauters

teaching assistant:

Maire Bowler

textbook:

Database System Concepts. 6th Edition
by *Silberschatz, Korth and Sudarshan*

slides:

QUB online, and www.db-book.com for textbook slides

introduction and course outlines

module outlines

course structure:

30 lectures/tutorials, 6 project presentations

7 practical sessions

lecture locations:

- Lecture: Mondays: 11am – noon, ECS02/0G/007
- Lecture: Tuesdays: 10 – 11am, ECS02/0G/007
- Lab Practical: Tuesdays: 1 – 3pm, ECS01/02/005, ECS01/02/014
- Lecture/Tutorial: Fridays: 11am – noon, ECS02/0G/007

corridor

~~office~~ hour on Tuesdays from 11am – noon, ECS1 lobby

introduction and course outlines

lab outline

week no.	Tuesday	topic	assessed?
2	6th Oct	MySQL and MySQL Workbench	no
3	13th Oct	SQL	no
4	20th Oct	More SQL	no
5	27th Oct	Assignment on SQL	yes, due 6th Nov
6			
7	10th Nov	Eclipse and JDBC	no
8	17th Nov	JDBC	no
9	24th Nov	Assignment on JDBC	yes, due 7th Dec
10	1st Dec	Project	
11	8th Dec	Project	

introduction and course outlines

module outlines

course assessment:

- 20% from 2 practical assignments + 1 class test
- 20% from group project and presentation
- 60% from final examination

do well on assignments/tests: keep up with course during the year makes project and final examination that much easier!

introduction and course outlines

module outlines

course assessment:

- 20% from 2 practical assignments + 1 class test
- 20% from group project and presentation
- 60% from final examination

we create teams/"companies"
each group will consist of 4–5 people

do well on assignments/tests: keep up with course during the year
makes project and final examination that much easier!

introduction and course outlines

quick glance: data models

data models in DBMS:

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

introduction and course outlines

quick glance: data models

data models in DBMS:

table name

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

introduction and course outlines

quick glance: data models

data models in DBMS:

table name

attribute

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

introduction and course outlines

quick glance: data models

data models in DBMS:

table name

attribute

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

tuple (or row / record)

introduction and course outlines

quick glance: data models

data models in DBMS:

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
1657	Kim	Comp. Sci.	[redacted]
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

teaches

instructor_ID	course_id	semester	year
1657	CSC7072	1	2015
	…	…	…

introduction and course outlines

quick glance: data models

data models in DBMS:

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
1657	Kim	Comp. Sci.	[redacted]
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

teaches

instructor_ID	course_id	semester	year
1657	CSC7072	1	2015
	…	…	…

introduction and course outlines

quick glance: Data Manipulation Language

data retrieval/modification in DBMS:

Data Manipulation Language (DML), *a.k.a.* query language

- procedural DML: like Java programming language
  tell what to get, and how to get it (*i.e.* control flow)
- declarative DML: like magic
  tell what to get, I will sort out how to get it

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
1657	Kim	Comp. Sci.	[redacted]
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

teaches

instructor_ID	course_id	semester	year
1657	CSC7072	1	2015
	…	…	…

introduction and course outlines

quick glance: Data Manipulation Language

data retrieval/modification in DBMS:

Data Manipulation Language (DML), *a.k.a.* query language

- procedural DML: like Java programming language
  tell what to get, and how to get it (*i.e.* control flow)
- declarative DML: like magic
  tell what to get, I will sort out how to get it

SQL

Structure Query Language

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
1657	Kim	Comp. Sci.	[redacted]
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

teaches

instructor_ID	course_id	semester	year
1657	CSC7072	1	2015
	…	…	…

introduction and course outlines

quick glance: Data Manipulation Language

data retrieval/modification in DBMS:

Data Manipulation Language (DML): SQL examples

```
SELECT dept_name
FROM instructor
```

dept_name
Comp. Sci.
Music
Physics

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
1657	Kim	Comp. Sci.	[redacted]
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

teaches

instructor_ID	course_id	semester	year
1657	CSC7072	1	2015
	…	…	…

introduction and course outlines

quick glance: Data Manipulation Language

data retrieval/modification in DBMS:

Data Manipulation Language (DML): SQL examples

```
SELECT name
    FROM instructor
        WHERE dept_name = 'Comp. Sci.'
```

name
Kim
Srinivasan

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
1657	Kim	Comp. Sci.	[redacted]
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

teaches

instructor_ID	course_id	semester	year
1657	CSC7072	1	2015
	…	…	…

introduction and course outlines

quick glance: Data Manipulation Language

data retrieval/modification in DBMS:

Data Manipulation Language (DML): SQL examples

```
SELECT name, course_id
    FROM instructor, teaches
        WHERE ID = instructor_ID
```

name	course_id
Kim	CSC7072
…	…

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
1657	Kim	Comp. Sci.	[redacted]
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

teaches

instructor_ID	course_id	semester	year
1657	CSC7072	1	2015
	…	…	…

introduction and course outlines

quick glance: Data Manipulation Language

data retrieval/modification in DBMS:

Data Manipulation Language (DML): SQL examples

```
SELECT name, course_id as 'code'
    FROM instructor as IN, teaches as T
        WHERE IN.ID = T.instructor_ID
```

name	code
Kim	CSC7072
…	…

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
1657	Kim	Comp. Sci.	[redacted]
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

teaches

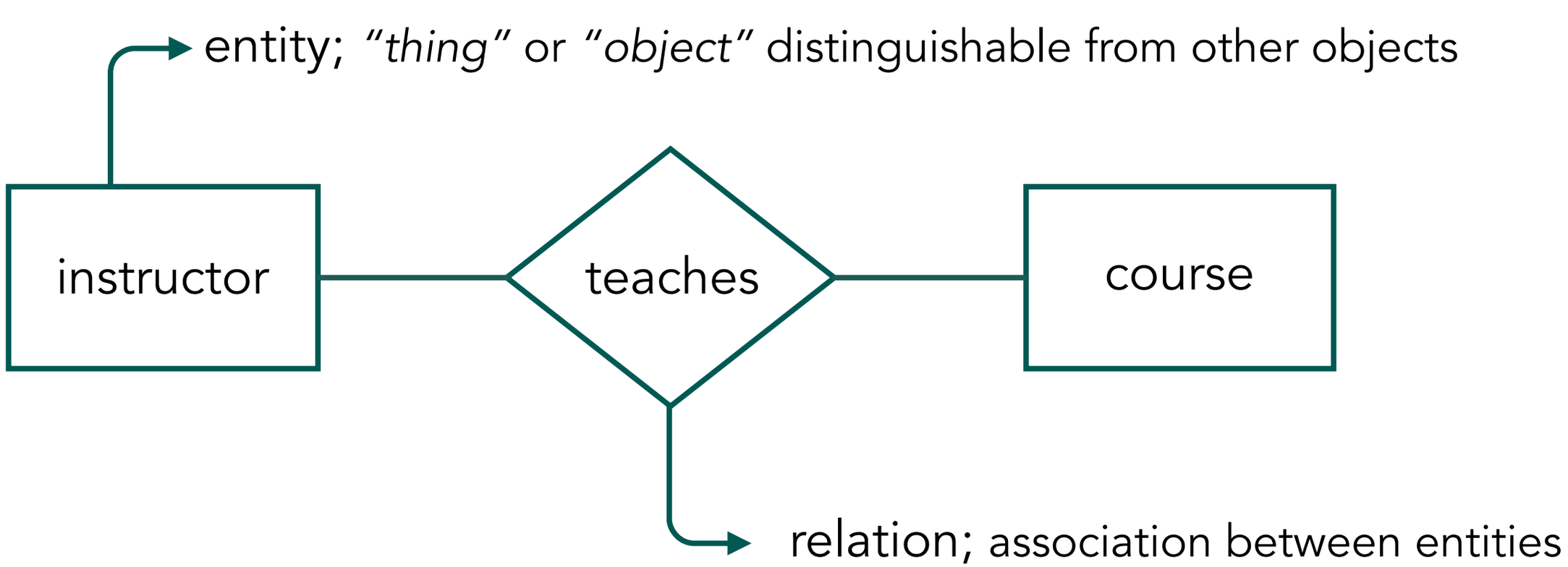
instructor_ID	course_id	semester	year
1657	CSC7072	1	2015
	…	…	…

introduction and course outlines

quick glance: database design using ER models and normalisation

database design (ER model) + database theory (normalisation):

given a problem specification, which tables do we need?



introduction and course outlines

quick glance: database design using ER models and normalisation

database design (ER model) + database theory (normalisation):

given a problem specification, which tables do we need?

- what is a normal form? i.e. what is an *ideal* table structure?
- is a database in normal form?
- how we can we transform a *bad* database into a *good* one?
  - inability to represent data
  - repetition of information

introduction and course outlines

quick glance: Data Definition Language

data definition in DBMS:

Data Definition Language (DDL): also SQL!

```
CREATE TABLE instructor (
    ID varchar(4),
    name varchar (20) not null,
    dept_name varchar (20),
    salary numeric (8,2),
    primary key (ID)
    )
```

instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
1657	Kim	Comp. Sci.	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000

+ impose constraints, ensure integrity, provide authorisation …

introduction and course outlines

quick glance: accessing databases and applications

accessing databases from programming languages

database applications

- transactions
  - how do we ensure atomic execution of changes?
    for example: person A transfers money to person B
    we do not want person A and B to be without money if
    the power goes down in the middle of transaction!
- data mining:
  - which products do we need to promote/advertise to which customers to maximise sales? *i.e.* find patters