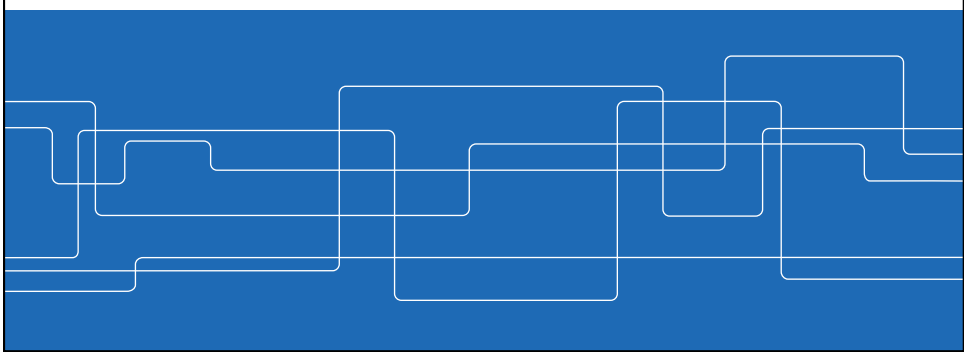
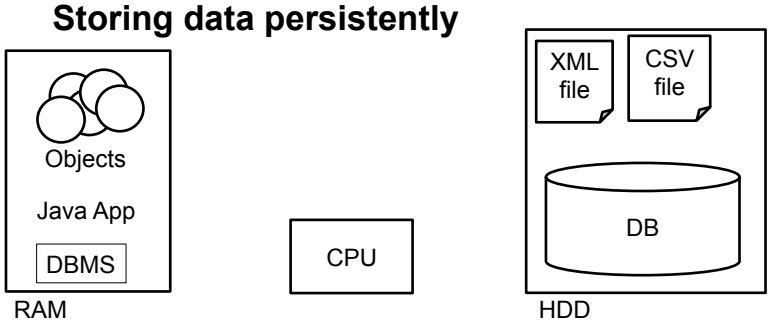
KTH ROYAL INSTITUTE
OF TECHNOLOGY

Lecture #8
Relational Database Systems



Contents

Storing data persistently



During execution, RAM is used to store our data
Files can be read and write for persistent storage
But what if we want to access the data in a more flexible way?
Reading single posts, adding data, removing data etc.

Databases
Database Management Systems (DBMS)

The idea of data storage developed gradually when computing capabilities grew – file storage was simply not enough

Various models to link and index that data were developed:

- Hierarchical storage (Tree-like struture)
- Network storage (cross references between data items)
- Relational storage (the winning model, used presently)

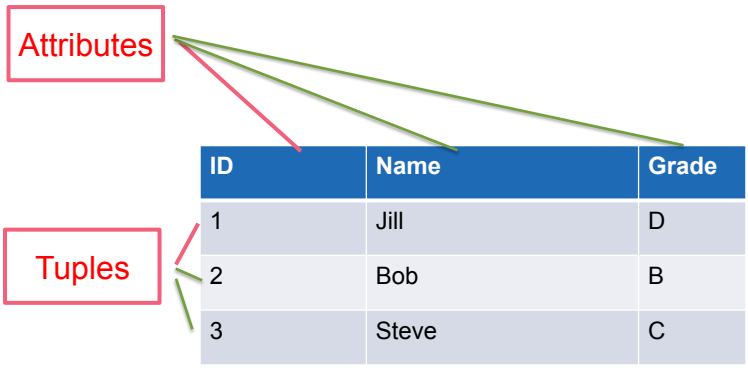
Relational data storage

Data is organised in tables of two dimensions
Rows & Columns

Tables are known as ”Relations”
Rows are ”Tuples”
Columns are ”Attributes”

ID	Name	Grade
1	Jill	D
2	Bob	B
3	Steve	C

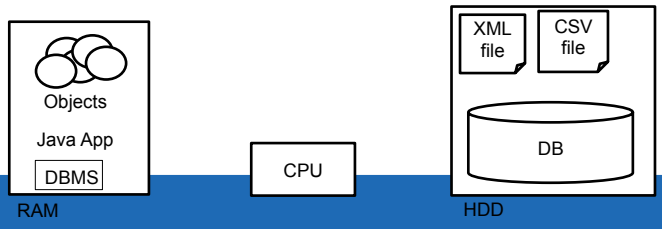
Cardinality & Degree



The cardinality of a Relation is its number of tuples (rows)
The degree of a Relation is its number of attributes (columns)

What are these tables really?

1. Data is stored on the computer's HDD as bits (of course)
2. The data is struuctred according to some scheme that is efficient for the disk and CPU's access to the data
3. When we people want to write a (Java) program to manipulate the data, we think of it, and access it in the form of tables
4. The DBMS program translate from the tables to actual data storage (which is logical to the CPU but not to us)



Contents

Entity Relationship Diagrams

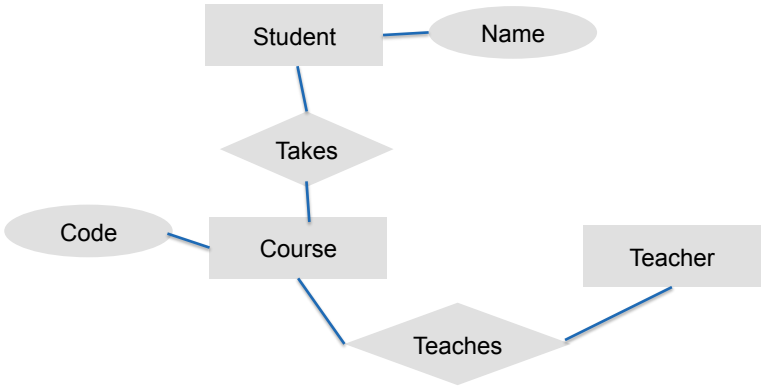
To Describe data and its relation E-R diagrams are used
They consist of:

Entities

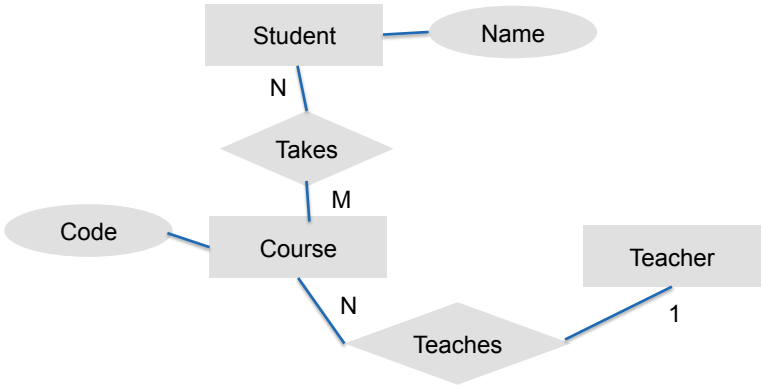
Attributes

Relationships

E-R Diagram example



Cardinality



Converting ER Diagrams to Tables

Convert all the Entities in the diagram to tables.

All single valued attributes of an entity is converted to a column of the table

Key attribute in the ER diagram becomes the Primary key of the table.

Declare the foreign key column, if applicable.

Convert Multi-value attributes into new tables

Convert Relations into Tables

Entity Relationship Diagrams as Tables

”Relations between Relations”

DEPARTMENT

No	Name
...	...

PROFESSORS

ID	Name	Dept-No	Courses
...	...	...	...

COURSES

No	Dept-No	Prof-ID	Unit
...	...	...	...

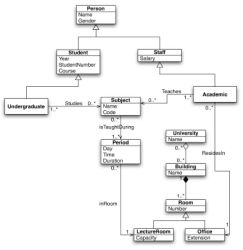
STUDENTS

ID	Name	Courses
...	...	...

By defining attributes as ”Keys” we can relate Tuples from different Relations to each other.

E-R diagrams vs. Class diagrams

But aren’t E-R diagrams the same as Class diagrams?



??

DEPARTMENT

No	Name
...	...

PROFESSORS

ID	Name	Dept-No	Courses
...	...	...	...

COURSES

No	Dept-No	Prof-ID	Unit
...	...	...	...

STUDENTS

ID	Name	Courses
...	...	...

In a way, they are very similar, but

- ER – only data, no methods
- ER – No OO concepts (inheritance, aggregation, etc.)
- Classes – No Keys

Contents

Good relations

Is this a good relation?

Part	Qt	Warehouse	Adress
Wheel	23	Building2	Main St 12
Wheel	12	Building1	Diagon Alley 3
Seat	9	Building1	Diagon Alley 3

Is this a good relation?

E-R Diagrams ”must” be Normalised

- Normalisation of E-R diagrams is like ”Good Programming Style” but for Data
- It enables more efficient access to data and more efficient storage
- Reduces the risk of error in data.

- In Theory 5 levels of Normality (or Normal forms) exist
  - 1st Normal form
  - 2nd Normal form
  - 3rd Normal Form
  - 4th Normal Form
  - 5th Normal Form

We stop here

First Normal Form

The First Normal is basic housekeeping.

- All Tuples in a Relation must have the same number of attributes.
- Or ”The degree of all Tuples” must be the same.

This borders on the obvious under the definition of a Relational Database, since this is the definition of a Relation

Second Normal Form

Only relevant when the keys are composite, i.e., consists of several attributes

To fulfill Second normal form non-key fields cannot have facts about a part of a key.

Warehouse	Part	Adress	Qt
Building2	Wheel	Main St 12	23
Building1	Wheel	Diagon Alley 3	12
Building1	Seat	Diagon Alley 3	9

Keys

Normalised to 2nd Normal Form

Warehouse	Part	Qt
Building2	Wheel	23
Building1	Wheel	12
Building1	Seat	9

Keys

Warehouse	Adress
Building1	Diagon Alley 3
Building2	Main St 12

Third Normal Form

In Third Normal Form, a non-key attribute must not hold information about another non-key attribute

Course	Professor	Office
EH2745	Nordström	Osquldas väg 10, floor 7
EH2751	Nordström	Osquldas väg 10, floor 7
EJ2301	Soulard	Teknikringen 33, floor 1
EG2200	Amelin	Reknikringen 35, floor 2

Key

Normalised to 3rd Normal form

Course	Professor
EH2745	Nordström
EH2751	Nordström
EJ2301	Soulard
EG2200	Amelin

Professor	Office
Nordström	Osquldas väg 10, floor 7
Amelin	Teknikringen 33, floor 2
Soulard	Teknikringen 33, floor 1

Contents

Tuple Relational Calculus

With the definitions (Relation, Tuple, Attribute) above we can define a number of basic operations on relations

```
Insert
Delete
Update
Select
Project
Join
Union
Intersection
Difference
```

Insert

Insert is a unary operation – it operates on a single Relation
It adds a Tuple to a Relation

ID	Name	Grade
1	Jill	D
2	Bob	B
3	Steve	C

ID	Name	Grade
1	Jill	D
2	Bob	B
3	Steve	C
4	Lars	A

```
Insert t in R
```

Delete

Delete is a unary operation – it operates on a single Relation
It deletes a Tuple fulfilling criteria from a Relation

ID	Name	Grade
1	Jill	D
2	Bob	B
3	Steve	C
4	Lars	A

ID	Name	Grade
1	Jill	D
2	Bob	B
4	Lars	A

```
Delete t where a=x from R
```

Update

Update is a unary operation – it operates on a single Relation
It modifies an attribute in Tuple fulfilling criteria in a Relation

ID	Name	Grade
1	Jill	D
2	Bob	B
4	Lars	A

ID	Name	Grade
1	Jill	D
2	Bob	B
4	Lars	E

```
Update t.a2=data where t.a1=x in R
```

Select

Select is a unary operation – it operates on a single Relation
The Select operation creates a new relation R2 from relation R1
The Tuples inR1 is a subset of R2

ID	Name	Grade
1	Jill	D
2	Bob	B
4	Lars	E

ID	Name	Grade
2	Bob	B
4	Lars	E

```
Select * from R1 where ID >1
```

Project

Project is a unary operation – it operates on a single Relation
The Project operation creates a new relation R2 from relation R1
The Attributes in R1 is a subset of R2

ID	Name	Grade
1	Jill	D
2	Bob	B
4	Lars	E

Name
Jill
Bob
Lars

```
Project Name from R1
```

Join

Join is a binary operation – it operates two Relations
The Join operation creates a new relation R3 from relations R1 & R2
Based on common attributes (keys)

Course	Professor
EH2745	Nordström
EH2751	Nordström
EJ2301	Soulard
EG2200	Amelin

X

Professor	Office
Nordström	Osquldas väg 10, floor 7
Amelin	Teknikringen 33, floor 2
Soulard	Teknikringen 33, floor 1

Not Normalised??

Course	Professor	Office
EH2745	Nordström	Osquldas väg 10, floor 7
EH2751	Nordström	Osquldas väg 10, floor 7
EJ2301	Soulard	Teknikringen 33, floor 1
EG2200	Amelin	Reknikringen 35, floor 2

Intermediate result for analysis

Union

A binary operation – it operates on two Relations R1 and R2

Creates a new relation R3 in which each tuple is either in R1, in the R2, or in both R1 and R2.

The two relations must have the same attributes.

Warehouse	Adress
Building3	King St 23
Building4	Queens Rd 2

+

Warehouse	Adress
Building1	Diagon Alley 3
Building2	Main St 12

Warehouse	Adress
Building1	Diagon Alley 3
Building2	Main St 12
Building3	King St 23
Building4	Queens Rd 2

Intersection

A binary operation – it operates on two Relations R1 and R2

Creates a new relation R3 in which each tuple is in both R1 and R2.

The two relations must have the same attributes.

Warehouse	Adress
Building1	Diagon Alley 3
Building4	Queens Rd 2

Warehouse	Adress
Building1	Diagon Alley 3
Building2	Main St 12

Warehouse	Adress
Building1	Diagon Alley 3

Difference

A binary operation – it operates on two Relations R1 and R2
Creates a new relation R3 in which each tuple is in R1 but not in R2.
The two relations must have the same attributes.

Warehouse	Adress
Building1	Diagon Alley 3
Building4	Queens Rd 2

-

Warehouse	Adress
Building1	Diagon Alley 3
Building2	Main St 12

Warehouse	Adress
Building4	Queens Road 2

Contents

Storing data
    Relational Database Systems
Entity Relationship diagrams
Normalisation of ER diagrams
Tuple Relational Calculus

Now you can try this with SQL in MySQL