Carnegie Mellon Univ. 
Dept. of Computer Science 
15-415/615 - DB Applications

C. Faloutsos - A. Pavlo 
Lecture#2: E-R diagrams

Administrivia

• Course url:
  – http://15415.courses.cs.cmu.edu/
• Course policies
  – http://15415.courses.cs.cmu.edu/fall2016/policies.html
• Foils in pps:
  – http://15415.courses.cs.cmu.edu/fall2016/slides-pps/

Course Topics

✔ Introduction to DBMSs
  • Data Models 
  • Query Language (SQL) 
  • Database Design 
  • Query Optimization & Indexing 
  • Transaction Management 
  • Advanced Topics 

Problem

• Develop an application for U.G. admin:
  – Student info 
  – Who-takes-what class 
  – Class rosters 
  – Transcripts 
• How do you proceed?
  – (Which role(s) are you playing?)
Database users

- ‘naive’ users
- casual users
- application programmers
- [DBA (Data base administrator)]

Casual users

```
select * from student
```

```
DBMS

data
and meta-data = catalog
```

App. programmers

- Authors of applications (like the ‘report generator’)

```
app. (eg., report generator)
```

```
DBMS

data
and meta-data = catalog
```

```
app. (eg., report generator)
```

```
DBMS

data
and meta-data = catalog
```
DB Administrator (DBA)

• Duties?

DBMS

and meta-data = catalog

data

DB Administrator (DBA)

• Duties?

DBMS

and meta-data = catalog

data

DB Administrator (DBA)

• schema definition (‘logical’ level)
• physical schema (storage structure, access methods)
• schema modifications
• granting authorizations
• integrity constraint specification

DB Administrator (DBA)

• Duties?

DBMS

and meta-data = catalog

data

Problem

• Develop an application for U.G. admin:
  – Student info
  – Who-takes-what class
  – Class rosters
  – Transcripts
• How do you proceed?
  – (Which role(s) are you playing?)
Database users

- 'naive’ users
- casual users
- application programmers
- [ DBA (Data base administrator)]

Database Design

- Requirements Analysis
- Conceptual Design
- Logical Design
- Schema Refinement
- Physical Design
- Security Design

Problem’

Maintain

- Develop an application for U.G. admin:
  - Student info
  - Who-takes-what class
  - Class rosters
  - Transcripts
- If you are the *new* DBA, what would you rather inherit:
This or this?

drop table if exists student;
create table student
  (ssn fixed,
   name char(20));
drop table if exists takes;
create table takes
  (ssn fixed,
   cid char(10),
   grade fixed);

True story

• Health insurance company
• Wants to catch (some of the abundant) fraud
• Schema:
  – patients, visit doctors, get medicine,
  – Doctors perform operations, …
  – Nurses monitor patients, …
  – etc etc
• Q: How many tables do you think it spans?

True story

• Schema:
  – patients, visit doctors, get medicine,
  – Doctors perform operations, …
  – Nurses monitor patients, …
  – etc etc
• Q: How many tables do you think it spans?
  10? 20? 30?
• A: 120 PAGES of schema
Motivation & upcoming conclusion:

- E-R diagrams are excellent documentation tools

```sql
drop table if exists student;
create table student
(ssn fixed,
name char(20));
drop table if exists takes;
create table takes
(ssn fixed,
cid char(10),
grade fixed);
```

Overview

- concepts
  - Entities
  - Relationships
  - Attributes
  - Specialization/Generalization
  - Aggregation
  - ER modeling questions

Tools

Entities (‘entity sets’)

Relationships (‘rel. sets’)

and mapping constraints

attributes

Example

Students, taking courses, offered by instructors; a course may have multiple sections; one instructor per section

nouns -> entity sets
verbs -> relationship sets
Example

Students, taking courses, offered by instructors; a course may have multiple sections; one instructor per section

nouns -> entity sets
verbs -> relationship sets

Example

Students, taking courses, offered by instructors; a course may have multiple sections; one instructor per section

nouns -> entity sets
verbs -> relationship sets

STUDENT name

INSTRUCTOR

primary key = unique identifier ->

but: sections of course (with different instructors)?

COURSE

e-id
e-name

INSTRUCTOR

STUDENT

name

...
Q: how to record that students take courses?

but: s-id is not unique... (see later)
Cardinalities

• 1 to 1 (example?)
• 1 to N
• N to M

Book’s notation:
Cardinalities

Book’s notation vs 1 to N notation

PERSON 1 owns N CAR
STUDENT N takes M SECTION

PERSON 1 owns N CAR
STUDENT N takes M SECTION

‘Total/partial’ participation

total, total

PERSON 1:1 owns 2:N CAR
STUDENT 2:M takes 2:M SECTION

PERSON 1:1 owns 0:N CAR
STUDENT 2:M takes 2:M SECTION
‘Total/partial’ participation

- Total, total
  - COUNTRY: has 1:1 CAPITAL
  - PERSON: owns 1:1 CAR
- Partial, total
  - PERSON: owns 0:N CAR
- Partial, total
  - STUDENT: takes 1:N SECTION

Subtle concept: Weak entities

- ‘section’ has no unique-id of its own! (?)

Is it ‘legal’?

NO! why not?
Weak entities

• ‘weak’ entities: if they need to borrow a unique id from a ‘strong entity - thick box.
• ‘c-id’ + ‘s-id’: unique id for SECTION
• partial key (eg., ‘s-id’) - dashed-underline
• identifying relationship (eg., ‘has’)

Weak entities

• Other example(s) of weak entities?

More details

• self-relationships - example?
More details

- self-relationships - example?

\[ \text{EMPLOYEE} \xrightarrow{1} \text{manages} \xrightarrow{N} \text{EMPLOYEE} \]

More details

- self-relationships - example?

\[ \text{FB user} \xrightarrow{??} \text{Has-friend} \xrightarrow{??} \text{FB user} \]

More details

- 3-way and k-way relationships?

\[ \text{EMPLOYEE} \xrightarrow{N} \text{uses} \xrightarrow{M} \text{TOOL} \xrightarrow{P} \text{PROJECT} \]

More details

- 3-way and k-way relationships? Rare, but possible:
More details

• 3-way and k-way relationships? Rare, but possible:

Other cases?

Overview

• concepts
  – Entities
  – Relationships
  – Attributes
  – Specialization/Generalization
  – Aggregation
  – ER modeling questions

More details - attributes

• key (or primary key): unique identifier
• underlined, in the ER diagram
• [not in textbook - FYI:
  – multivalued or set-valued attributes (eg., 'dependents' for
    EMPLOYEE)
  – derived attributes (eg., 15% tip)
  ]
Overview

- concepts
  - Entities
  - Relationships
  - Attributes
  - Specialization/Generalization
  - Aggregation
  - ER modeling questions

Specialization

- eg., students: part time (#credit-hours) and full time (major)

Observations

- Generalization: exact reverse of ‘specialization’
- attribute inheritance
- could have many levels of an IS-A hierarchy

More details

- Overlap constraints
- Covering constraints
More details

- Overlap constraints
  - can an entity belong to both ‘B’ and ‘C’?
- Covering constraints
  - can an ‘A’ entity belong to neither ‘B’ nor ‘C’?

More details

- Overlap constraints - examples?

No overlap

A

B

C

with overlap

A

B

C

More details

- Covering constraints - examples?

Total coverage

A

B

C

Partial coverage

A

B

C

Overview

- concepts
  - Entities
  - Relationships
  - Attributes
  - Specialization/Generalization
  - Aggregation
  - ER modeling questions
Aggregation

- computer model (w/ CPU and HD)
- and Maker (e.g., Dell, HP)

![Diagram showing computer model with CPU and HD connected to Maker](image1)

Aggregation

- treat a relationship as an entity
- used to express a relationship among relationships

![Diagram showing CPU and HD connected to Maker](image2)

Overview

- concepts
  - Entities
  - Relationships
  - Attributes
  - Specialization/Generalization
  - Aggregation
- ER modeling questions

![Diagram showing relationships between concepts](image3)

Conceptual design

- Entity vs attribute
- Entity vs relationship
- Binary or ternary relationships?
- Aggregation?
Entity vs. attribute

• Entity EMPLOYEE (w/ emp#, name, job_code, ...)
• Q: How about ‘spouse’ - entity or attribute?
• Q: How about ‘dependents’?

• Entity EMPLOYEE (w/ emp#, name, job_code, ...)
• Q: How about ‘spouse’ - entity or attribute?
• A: probably, ‘attribute’ is enough
• Q: How about ‘dependents’?
• A: Entity - we may have many dependents

Entity vs. Relationship

STUDENT

M
N
takes

SECTION

TAKES

N
M

STUDENT

 Binary vs Ternary Relationships

• usually, binary relationships are ‘cleaner’:
**Binary vs. Ternary Relationships**

If each policy is owned by just 1 employee:

**Bad design**

Key constraint on Policies would mean policy can only cover 1 dependent!

Better design

What are the additional constraints in the 2nd diagram?
Binary vs Ternary Rel.

• But sometimes ternary rel. can not be replaced by a set of binary rel’s:

Binary vs. Ternary Relationships (Contd.)

- S “can-supply” P, D “needs” P, and D “deals-with” S does not imply that D has agreed to buy P from S.
- How do we record qty?

Not in textbook: in practice, often:
Binary vs. Ternary Relationships (Contd.)

Not in textbook:
in practice, often:

```
<table>
<thead>
<tr>
<th>qty</th>
<th>c-id</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

```
Parts  Contract  Departments
Suppliers
```

Ternary vs. aggregation

- use aggregation, if we want to attach a relationship to a relationship
- (see book for example)
- (in practice, again we create a unique-id and resort to binary relationships)

Binary vs. Ternary Relationships (Contd.)

Not in textbook:
in practice, often:

```
<table>
<thead>
<tr>
<th>qty</th>
<th>c-id</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

```
Parts  Contract  Departments
Suppliers
```

Ternary vs. aggregation

- How would you handle this case?

```
CPU  HD  MAKER
```

```
M  N
```
Ternary vs. aggregation

- How would you handle this case?

```
COMP. MODEL
CPU  HD  MAKER
```

Summary

- E-R Diagrams: a powerful, user-friendly tool for data modeling:
  - Entities (strong, weak)
  - Attributes (primary keys, discriminators, derived, multivalued)
  - Relationships (1:1, 1:N, N:M; multi-way)
  - Generalization/Specialization; Aggregation
Summary

• E-R Diagrams: a powerful, user-friendly tool for data modeling:
  – Entities (strong, weak)
  – Attributes (primary keys, discriminators, derived, multivalued)
  – Relationships (1:1, 1:N, N:M; multi-way)
  – Generalization/Specialization; Aggregation

Summary - cont’d

- (strong) entity set
- weak entity set
- relationship set
- identifying rel. set for weak entity
- attribute
- primary key
- partial key
- IS-A
- aggregation

(cardinalities)

N M

1:1:1' h' cardinalities with limits

(not in textbook - FYI)