CS 377
Database Systems
Entity-Relationship Model

Liyue Fan
Department of Mathematics and Computer Science
Emory University
ER Model


ER Model Concepts

Entities
Relationships

ER Diagrams
Example Database

Each course at Emory has a unique course number. Each course can have more than one sections and can be offered in different semesters/years. Each section is taught by one and only one professor but may enroll many students. Professors may advise more than one students but each student can only have one adviser. Some professors(Chair, Dean) supervise other professors.
Entities and Attributes

**Entities** are specific objects or things in the mini-world that are represented in the database.

E.g. STUDENT John Smith, the CS377 COURSE

**Attributes** are properties used to describe an entity.

E.g. Name, Student ID, Address, Sex, BirthDate of an STUDENT entity

Each attribute has a *value set* (or data type) associated with it

E.g. integer, string, subrange, enumerated type, …

A specific entity will have a value for each of its attributes.

E.g. A specific student entity may have Name='John Smith', Sid='123456789', Address = '731, Fondren, Houston, TX', Sex='M', BirthDate='09-JAN-90'
Types of Attributes

Simple - Each entity has a single atomic value for the attribute.
   E. g Sid or Sex.

Composite - The attribute may be composed of several components.
   E.g. Address (Apt#, House#, Street, City, State, ZipCode, Country) or
   Name (FirstName, MiddleName, LastName).

Multi-valued - An entity may have multiple values for that attribute.
   E.g. Color of a CAR or PreviousDegrees of a STUDENT. Denoted as
   {Color} or {PreviousDegrees}.

Complex - Composite and multi-valued attributes may be nested
arbitrarily to any number of levels although this is rare.
   E.g. PreviousDegrees of a STUDENT is a composite multi-valued attribute
   denoted by {PreviousDegrees (College, Year, Degree, Field)}.

Derived attributes - values of these attributes can be derived from
other attributes.
   E.g.. age - can be derived from "birth date".
Key Attributes

An attribute of an entity type for which each entity must have a unique value is called a **key attribute** of the entity type.

E.g. id of STUDENT.

A key attribute may be composite.

E.g. VehicleTagNumber is a key of the CAR entity type with components (Number, State).

An entity type may have more than one key.

E.g. the CAR entity type may have two keys: VehicleIdentificationNumber (popularly called VIN) and VehicleTagNumber (Number, State).
Entity Type and Entity Set

Entities with the same basic attributes are grouped or typed into an entity type.

E.g. the STUDENT entity type or the COURSE entity type.

Entity set is a collection of an entity type at a specific point of time
ER-DIAGRAM NOTATION

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>E₁</td>
<td>ENTITY TYPE</td>
</tr>
<tr>
<td>E₂</td>
<td>WEAK ENTITY TYPE</td>
</tr>
<tr>
<td>R</td>
<td>RELATIONSHIP TYPE</td>
</tr>
<tr>
<td>N</td>
<td>IDENTIFYING RELATIONSHIP TYPE</td>
</tr>
<tr>
<td>(min,max)</td>
<td>ATTRIBUTE</td>
</tr>
<tr>
<td></td>
<td>KEY ATTRIBUTE</td>
</tr>
<tr>
<td></td>
<td>MULTIVALUED ATTRIBUTE</td>
</tr>
<tr>
<td></td>
<td>COMPOSITE ATTRIBUTE</td>
</tr>
<tr>
<td></td>
<td>DERIVED ATTRIBUTE</td>
</tr>
<tr>
<td></td>
<td>TOTAL PARTICIPATION OF E₂ IN R</td>
</tr>
<tr>
<td></td>
<td>CARDINALITY RATIO 1:N FOR E₁:E₂ IN R</td>
</tr>
<tr>
<td></td>
<td>STRUCTURAL CONSTRAINT (min, max) ON PARTICIPATION OF E IN R</td>
</tr>
</tbody>
</table>
Design of the University Database

Step 1. Identify entity types
Step 2. Identify attributes for each entity type
To be continued …
A relationship relates two or more distinct entities with a specific meaning.
   E.g. STUDENT John Smith takes the CS377 SECTION 000

Relationships of the same type are grouped or typed into a relationship type.
   E.g. the TAKE relationship type in which STUDENTs and SECTIONs participate

More than one relationship type can exist with the same participating entity types.
Degree of Relationship

The **degree** of a relationship type is the number of participating entity types.

Relationship types of degree 2 are called **binary**

E.g. Both TAKE and TEACH are binary relationships.

Relationship types of degree 3 are called **ternary**
and of degree n are called **n-ary**

An n-ary relationship equivalent to n binary relationships?
Recursive Relationships

Relationships can be **recursive**: Both participations are same entity type in different roles.

E.g. SUPERVISE relationship between PROFESSOR (in role of chair or dean) and (another) PROFESSOR (in role of instructor).
Attributes of Relationship types

A relationship type can have attributes

E.g. HoursPerWeek of ADVISE describing the number of hours per week that an PROFESSOR spends on advising a STUDENT.
Weak Entity Types

A **weak entity** is an entity that does not have a key attribute and participates in an **identifying relationship type** with an **owner or identifying entity type**.

Entities are identified by the combination of:

- A **partial key** of the weak entity type
- The particular entity they are related to in the identifying entity type

**Example:**

A SECTION entity is identified by the section number *and* the specific COURSE that the section is related to. SECTION is a weak entity type with COURSE as its identifying entity type via the identifying relationship type SECTIONS.
**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 E₂ IN R

CARDINALITY RATIO 1:N FOR E₁:E₂ IN R

STRUCTURAL CONSTRAINT (min, max) ON PARTICIPATION OF E IN R
Structural Constraints on Relationship Types

Maximum Cardinality
   One-to-one (1:1)
   One-to-many (1:N) or Many-to-one (N:1)
   Many-to-many

Minimum Cardinality (also called participation constraint or existence dependency constraints)
   zero (partial participation, optional participation, not existence-dependent)
   one or more (total participation, mandatory, existence-dependent)
Structural Constraints for Relationships - ER Diagram

**Cardinality ratio** (of a binary relationship): 1:1, 1:N, N:1, or M:N shown by placing appropriate number on the link.

**Participation constraint** (on each participating entity type): total (called *existence dependency*) shown by double lining the link.

Alternative (min, max) notation specifying that each entity e in E participates in *at least* min and *at most* max relationship instances in R

Default (no constraint): min=0, max=n

E.g. A student can have at most one adviser. A section must be taught, and can only be taught by one professor.

- Specify (0,1) for participation of STUDENT in ADVISE
- Specify (1,1) for participation of SECTION in TEACH
# Database Design Tools

<table>
<thead>
<tr>
<th>COMPANY</th>
<th>TOOL</th>
<th>FUNCTIONALITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Embarcadero Technologies</td>
<td>ER Studio</td>
<td>Database Modeling in ER and IDEF1X</td>
</tr>
<tr>
<td></td>
<td>DB Artisan</td>
<td>Database administration and space and security management</td>
</tr>
<tr>
<td>Oracle</td>
<td>Developer 2000 and Designer 2000</td>
<td>Database modeling, application development</td>
</tr>
<tr>
<td>Popkin Software</td>
<td>System Architect 2001</td>
<td>Data modeling, object modeling, process modeling, structured analysis/design</td>
</tr>
<tr>
<td>Platinum Technology</td>
<td>Platinum Enterprice Modeling Suite: Erwin, BPWin, Paradigm Plus</td>
<td>Data, process, and business component modeling</td>
</tr>
<tr>
<td>Persistence Inc.</td>
<td>Pwertier</td>
<td>Mapping from O-O to relational model</td>
</tr>
<tr>
<td>Rational</td>
<td>Rational Rose</td>
<td>Modeling in UML and application generation in C++ and JAVA</td>
</tr>
<tr>
<td>Rogue Ware</td>
<td>RW Metro</td>
<td>Mapping from O-O to relational model</td>
</tr>
<tr>
<td>Resolution Ltd.</td>
<td>Xcase</td>
<td>Conceptual modeling up to code maintenance</td>
</tr>
<tr>
<td>Sybase</td>
<td>Enterprise Application Suite</td>
<td>Data modeling, business logic modeling</td>
</tr>
<tr>
<td>Visio</td>
<td>Visio Enterprise</td>
<td>Data modeling, design and reengineering Visual Basic and Visual C++</td>
</tr>
</tbody>
</table>