Modeling: Entity-Relationship Diagrams
Scenario

• http://www.imdb.com wants to store information about movies and has chosen you to help them
• Four steps:
  – Requirements Analysis: Discover what information needs to be stored, how the stored information will be used, etc. Taught in course on system analysis and design
  – Conceptual Database Design: High level description of data to be stored (ER model)
  – Logical Database Design: Translation of ER diagram to a relational database schema (description of tables)
  – Physical Database Design: Done by the DB system
Requirements

• For actors and directors, we want to store their name, a unique identification number, address and birthday (why not age?)
• For actors, we also want to store a photograph
• For films, we want to store the title, year of production and type (thriller, comedy, etc.)
• We want to know who directed and who acted in each film. Every film has one director. We store the salary of each actor for each film
• Etc…
ER-Diagrams: General Information

- ER-diagrams are a formalism to model real-world scenarios.
- There are many versions of ER-diagrams that differ both in their appearance and in their meaning.
  - We will use the version appearing in the book *Database Systems: The Complete Book*.
- ER-diagrams have a formal semantics (meaning) that must be thoroughly understood, in order to create correct diagrams.
- **Goal** of modeling is to translate informal requirements to a precise diagram. This diagram can then be translated into the desired **data model**, to allow data to be stored in a database.
Basic Concepts:
Entities, Attributes, Relationships
Entities, Entity Sets

- **Entity** (ישות): An object in the world that can be distinguished from other objects
  - Examples of entities:
  - Examples of things that are not entities:

- **Entity set** (קבוצת ישויות): A set of similar entities
  - Examples of entity sets:

✍ Entity sets are drawn as rectangles
• **Attributes (תכונות):** Used to describe entities
  – All entities in the set have the same attributes
  – A minimal set of attributes that uniquely identify an entity is called a **key**
  – An attribute contains a single piece of information (and not a list of data)
Attributes (2)

• Examples of attributes:

• Examples of things that cannot be attributes:

- Attributes are drawn using ovals
- The names of the attributes which make up a key are underlined
Example

id
name
Actor
birthday
address
Another Option for a Key?

Actor

- id
- name
- birthday
- address
Another Option for a Key?

id

name

Actor

birthday

address
• **Relationship** ( קשר): Association among two or more entities
  – Examples of Relationships:

• **Relationship Set** ( קבוצת קשרים): Set of similar relationships
  – Examples of Relationship sets:

❗ Relationship sets are drawn using diamonds
Example
Recursive Relationships

- An entity set can participate more than once in a relationship
- In this case, we add a description of the role to the ER-diagram
**n-ary Relationship**

- An *n*-ary relationship set \( R \) involves exactly *n* entity sets: \( E_1, \ldots, E_n \).
- Each relationship in \( R \) involves exactly *n* entities: \( e_1 \) in \( E_1 \), \( \ldots \), \( e_n \) in \( E_n \).
- Formally, \( R \subseteq E_1 \times \cdots \times E_n \).
Example

• Suppose that there are:
  – 2 Actors
  – 3 Directors
  – 4 Film

How many triples can be in the relationship set “Directed”?
How many pairs can be in the relationship set “Directed”?
Multiplicity of Relationships
A member of E may be connected by R to any number of members from F, and vice versa.

This is called a many-many relationship.
Many-to-Many

✓ A film is directed by any number of directors
✓ A director can direct any number of films
• By adding arrows to the diagram, we can indicate constraints on the relationship

• An arrow towards F indicates that:
  – A member of E may be connected by R to at most one members from F
  – (Still, a member of F may be connected by R to any number of members from E)

• This is called a **many-one relationship**
**One-to-Many**

- A film is directed by at most one director
- A director can direct any number of films
Multiplicity of Relationships

- An arrow towards F and towards E indicates that:
  - A member of E may be connected by R to at most one member from F
  - A member of F may be connected by R to at most one member from E

- This is called a one-one relationship
One-to-One

- A film is directed by at most one director
- A director can direct at most one film
Example

Where would you put the arrow?
Another Example

Where would you put the arrow?

Diagram:
- Person (id, name, age)
- FatherOf (father, child)

Arrows: father points to Person, child points to FatherOf.
For any $1 \leq i \leq m$

For any tuple of entities $e_1, \ldots, e_n, f_{1}, \ldots, f_{i-1}, f_{i+1}, \ldots, f_m$

there is at most one $f_i$, such that $e_1, \ldots, e_n, f_1, \ldots, f_m$

are connected by $R$
Suppose that there are
• a entities in A
• b entities in B
• c entities in C
• d entities in D

What is the maximum number of 4-tuples in R?
What is the minimum number of 4-tuples in R?
What does this mean?

Note that for many reasons, this is a bad modeling.
Multiplicities in Multiway Relationships

Note that for many reasons, this is a bad modeling

Each pair of a man and a woman can have at most one child
Referential Integrity and Degree Constraints
Referential Integrity

- So far, we can say that an entity participates at most one time, but cannot require it to participate at least one time.

- The rounded arrow above indicates that each entity in E must participate exactly one time in an R-relationship with an entity in F.
We can attach a bounding number to edges to indicate limits on the number of entities that can be connected to a single entity via a relationship set. In the example above, a move has at most 10 stars.

Note: a regular arrow is the constraint <=1.

Note: a rounded arrow is the constraint =1.
Example (1)

What does this mean?

Diagram:
- Student
  - id
  - name
  - age
- StudiesAt
- University
  - name
Example (2)

What does this mean?

Note that for many reasons, this is a terrible modeling.
Relationship Sets with Attributes
Where does the salary attribute belong?
• The entities in a relationship set must identify the relationship
• Attributes of the relationship set cannot be used for identification!
• Suppose we wanted to store the role of an actor in a film.
  – How should we store the role of the actor?
  – How would we store information about a person who acted in one film in several roles?
Subclasses
ISA Hierarchies

ISA Relationships: Defines a hierarchy between entity sets

- ISA is similar to inheritance

ISA relationships are drawn as a triangle with the word ISA inside it. The "super entity-set" is above the triangle and the "sub entity-sets" are below.
Implications of an ISA Relationship

- Every entity in B or in C belongs to A
- There may be entities in A that do not belong to B or to C
- There may be entities that belong to both B and C
What are the keys of:
1. Movie Person
2. Actor
3. Director
Is this good method of modeling data for the "רבנות" database on marriage?

How can you fix it?
Weak Entity Sets
Intuition

• Sometimes, entities cannot be identified by their own attributes.

• To identify such an entity, we need information about a “supporting relationship”
  – Example: Given a bank account number, you cannot identify the actual bank account. For identification, you also have to know the name of the bank.
Example

- A 플וגה has a letter (א, ב, ...) ...
- To uniquely identify the 플וגה, one must know which גדוד it belongs to
- 플וגה is a weak entity set
- The relationship set שייכת ל is the supporting relationship for 플וגה וגדוד

גזרת
מספר גדוד
שייכת ל
אות 플וגה
פלוגה
Notation

• Weak entity set has a double line
• Supporting relationships have double lines
• Rounded arrow pointing into the identifying entity sets
Example (1)

- Keys:
- מספר גדוד: גדוד
- אות פלוגה, מספר גדוד: פלוגה
- מספר מחלקה, אות פלוגה, מספר גדוד מחלקה
Example (2)

- Same award can be given by several organizations ("Academy award for Best Actor 2007"),
- A year, award name and organization name uniquely define an award
- Weak entity set can participate in additional (non-supporting) relationships
Example (3)

- Awards are now identified by organization and country
- Same award can be given by same organization in different countries ("Academy award for Best Actor Israeli 2007")
- Weak identity set has 2 supporting relationships
- What is identifying key for award?
Design Principles
Faithfulness

• The design should be accurate to the specifications

  ![Diagram](diagram.png)

• This is ok *only* if each actor has a set salary, regardless of all movies
Avoiding Redundancy

- The design should not model the same information in multiple ways

- Leads to fact repetition
- Leads to inconsistencies
Simplicity Counts

- Avoid introducing elements that are not needed.
- If we never need to store information about people that are not movie people, don’t put it in the diagram.
Picking the Right Kind of Element

• The bottom diagram is sufficient if Studio has no attributes other than name.
Summary

• Given a set of requirements, to translate the requirements into a diagram:
  – Identify the entity sets
  – Determine if there are hierarchies (ISA or weak relationships) among entity sets
  – Identify the relationship sets
  – Identify the attributes
  – Determine constraints on relationship participation
The Relational Model
Data Models

• A **data model** is a notation for describing data
  – Conceptual structure of the data
  – Operations on the data
  – Constraints on the data

• In this course we focus on the **relational data model**
The basic element of the relational model is a **relation** (which is similar to a table)

A relation has a **schema**, consisting of a

- **Name**
- **List of attributes**, possibly with domains

A relation may also have an **instance**, which is a set of **tuples** (i.e., rows) in the relation
• **Schema**: Movies(title, year, length, genre)

• **Relation name**: Movies

• **Attributes**: title, year, length, genre

• **Possible tuple instance**
  – (“Follow that Bird”, 1985, 90, children)

• **Scheme with domains**:

<table>
<thead>
<tr>
<th>Title</th>
<th>Year</th>
<th>Length</th>
<th>Genre</th>
</tr>
</thead>
<tbody>
<tr>
<td>Follow…</td>
<td>1985</td>
<td>90</td>
<td>children</td>
</tr>
<tr>
<td>Who …</td>
<td>1987</td>
<td>90</td>
<td>mystery</td>
</tr>
</tbody>
</table>
Operations on the Data

• Relational algebra
  – Selection, projection, union, minus, join

• Stay tuned… Discussed in detail next week
Constraints on the Data

• We discuss complex constraints later on in the course.

• For now, we introduce **key constraints**.

• A set of attributes forms a **key** for a relation if there cannot be 2 different tuples with the same values for all attributes of the key.
  – Noted with underline.

• Examples:
  – Movies(\texttt{title}, \texttt{year}, \texttt{length}, \texttt{genre})
  – Actor(\texttt{teudatZehut}, \texttt{name}, \texttt{address})
A Step Closer

- Once we have a set of relational schemas in the relational model, we are a step closer to storing data in a DBMS
  - A DBMS has a **data definition language (DDL)**, used to define **tables** in the database
  - Once we have decided on the relational schemas, these can be directly translated into the database using the DDL
ER Diagrams to Relational Schemas
**General Principles**

- When converting ER diagrams to Relations, we should:
  - Reduce duplicate information
  - Constrain as tightly as possible

- Notes:
  - Some scenarios can be represented in different ways.
  - Sometimes we will not be able to fully represent constraints, or will be forced to represent information more than once.
Entity Set Translation

General Rule:

- Create a relation with the name of the Entity.
- There is a column for each attribute.
- The key in the diagram is the primary key of the relation.

Actor (id, name, birthday, address)
**General Rule:**
- Create a table with the name of the relationship set
- Relationship table attributes: its own attributes (salary) + all keys of the relating entities (title, id)
- What is the primary key of the table?
- Note: Do not define two attributes with the same name – instead rename one of them in the schema
What are all the relations created for this diagram?
Translating relationships (one-to-many): Option 1

Option 1:
Same as without key constraints (3 tables), except that the primary key of Directed is now title (why?)
Translating relationships (one-to-many): Option 2

Option 2:

- Do not create a table for the relationship
- Add information columns that would have been in the relationship's relation to the relation of the entity which does not have the incoming arrow
Translating Weak Entity Sets

- Key of relation for weak entity set includes its own keys and the keys of its supporting entity sets
- No relation is created for the supporting relationship sets

- Example: Translate diagram on the left to relations
Translating ISA: E/R Style Conversion

- A relation for each entity set. An entity may appear in more than one relation

MoviePerson(id, address, name)
Actor(id, picture)
Director(id)
A relation for each possible combinations of how entities may appear in the entity sets.

MoviePerson(id, address, name)
MoviePersonActor(id, address, name, picture)
MoviePersonDirector(id, address, name)
MoviePersonActorDirector(id, address, name, picture)
Translating ISA: Null Value Approach

- A single relation for containing all values
- Possible only if NULL values (i.e., missing values) are allowed
  - not in the pure relational model

MoviePerson(id, address, name, picture)