ER to Relational Mapping

Logical DB Design: ER to Relational

• Entity sets to tables.

CREATE TABLE Employees (ssn CHAR(11), name CHAR(20), lot INTEGER, PRIMARY KEY (ssn))

<table>
<thead>
<tr>
<th>ssn</th>
<th>name</th>
<th>lot</th>
</tr>
</thead>
<tbody>
<tr>
<td>123-22-3666</td>
<td>Attishoo</td>
<td>48</td>
</tr>
<tr>
<td>231-31-5368</td>
<td>Smiley</td>
<td>22</td>
</tr>
<tr>
<td>131-24-3650</td>
<td>Smethurst</td>
<td>35</td>
</tr>
</tbody>
</table>

CREATE TABLE Departments (did INTEGER, dname CHAR(20), budget REAL, PRIMARY KEY (did))

CREATE TABLE Manages (ssn CHAR(11), did INTEGER, since DATE, PRIMARY KEY (did), FOREIGN KEY (ssn) REFERENCES Employees, FOREIGN KEY (did) REFERENCES Departments)

<table>
<thead>
<tr>
<th>ssn</th>
<th>did</th>
<th>since</th>
</tr>
</thead>
<tbody>
<tr>
<td>123-22-3666</td>
<td>51</td>
<td>1/1/91</td>
</tr>
<tr>
<td>123-22-3666</td>
<td>56</td>
<td>3/3/93</td>
</tr>
<tr>
<td>231-31-5368</td>
<td>51</td>
<td>2/2/92</td>
</tr>
</tbody>
</table>

CREATE TABLE Dept_Mgr (did INTEGER, dname CHAR(20), budget REAL, ssn CHAR(11), since DATE, PRIMARY KEY (did), FOREIGN KEY (ssn) REFERENCES Employees)

Relationship Sets to Tables

• In translating a many-to-many relationship set to a relation, attributes of the relation must include:
  – Keys for each participating entity set (as foreign keys).
  – This set of attributes forms a superkey for the relation.
  – All descriptive attributes.

CREATE TABLE Works_In (ssn CHAR(11), did INTEGER, since DATE, PRIMARY KEY (ssn, did), FOREIGN KEY (ssn) REFERENCES Employees, FOREIGN KEY (did) REFERENCES Departments)

<table>
<thead>
<tr>
<th>ssn</th>
<th>did</th>
<th>since</th>
</tr>
</thead>
<tbody>
<tr>
<td>123-22-3666</td>
<td>51</td>
<td>1/1/91</td>
</tr>
<tr>
<td>123-22-3666</td>
<td>56</td>
<td>3/3/93</td>
</tr>
<tr>
<td>231-31-5368</td>
<td>51</td>
<td>2/2/92</td>
</tr>
</tbody>
</table>

Review: Key Constraints

• Each dept has at most one manager, according to the key constraint on Manages.

Review: Participation Constraints

• Does every department have a manager?
  – If so, this is a participation constraint: the participation of Departments in Manages is said to be total (vs. partial).
  Every did value in Departments table must appear in a row of the Manages table (with a non-null ssn value!)

CREATE TABLE Managers (did INTEGER, since DATE, PRIMARY KEY (did), FOREIGN KEY (did) REFERENCES Departments)

CREATE TABLE Dept_Mgr (did INTEGER, dname CHAR(20), budget REAL, ssn CHAR(11), since DATE, PRIMARY KEY (did), FOREIGN KEY (ssn) REFERENCES Employees)
Participation Constraints in SQL

- We can capture participation constraints involving one entity set in a binary relationship, but little else (without resorting to CHECK constraints).

```
CREATE TABLE Dept_Mgr (  
  did INTEGER,  
  dname CHAR(20),  
  budget REAL,  
  ssn CHAR(11) NOT NULL,  
  PRIMARY KEY (did),  
  FOREIGN KEY (ssn) REFERENCES Employees,  
  ON DELETE NO ACTION)
```

Translating Weak Entity Sets

- Weak entity set and identifying relationship set are translated into a single table.
  - When the owner entity is deleted, all owned weak entities must also be deleted.

```
CREATE TABLE Dep_Policy (  
  pname CHAR(20),  
  age INTEGER,  
  cost REAL,  
  ssn CHAR(11) NOT NULL,  
  PRIMARY KEY (pname, ssn),  
  FOREIGN KEY (ssn) REFERENCES Employees,  
  ON DELETE CASCADE)
```

Translating ISA Hierarchies to Relations

- General approach:
  - 3 relations: Employees, Hourly_Emps and Contract_Emps.
  - Hourly_Emps: Every employee is recorded in Employees. For hourly emp, extra info recorded in Hourly_Emps (hourly_wages, hours_worked, ssn); must delete Hourly_Emps tuple if referenced Employees tuple is deleted.
  - Queries involving all employees easy, those involving just Hourly_Emps require a join to get some attributes.
  - Alternative: Just Hourly_Emps and Contract_Emps.
    - Hourly_Emps: emp, name, lot, hourly_wages, hours_worked.
    - Each employee must be in one of these two subclasses.

Review: Weak Entities

- A weak entity can be identified uniquely only by considering the primary key of another (owner) entity.
  - Owner entity set and weak entity set must participate in a one-to-many relationship set (1 owner, many weak entities).
  - Weak entity must have total participation in this identifying relationship set.

Review: ISA Hierarchies

- As in C++, or other PLs, attributes are inherited.
  - if we declare A ISA B, every A entity is also considered to be a B entity.
  - Overlap constraints: Can Joe be an Hourly_Emps as well as a Contract_Emps entity? (Allowed/disallowed)
  - Covering constraints: Does every Employees entity also have to be an Hourly_Emps or a Contract_Emps entity? (Yes/no)

Review: Binary vs. Ternary Rel’ns

- If each policy is owned by just 1 employee:
  - Key constraint on Policies would mean policy can only cover 1 dependent!

Better design
**Binary vs. Ternary Relationships (Contd.)**

- The key constraints allow us to combine Purchaser with Policies and Beneficiary with Dependents.
- Participation constraints lead to NOT NULL constraints.

```
CREATE TABLE Policies (
    policyid INTEGER,
    cost REAL,
    ssn CHAR(11) NOT NULL,
    PRIMARY KEY (policyid),
    FOREIGN KEY (ssn) REFERENCES Employees,
    ON DELETE CASCADE
)

CREATE TABLE Dependents (
    pname CHAR(20),
    age INTEGER,
    policyid INTEGER,
    PRIMARY KEY (pname, policyid),
    FOREIGN KEY (policyid) REFERENCES Policies,
    ON DELETE CASCADE
)
```

**ER Model Summary**

- Usually easier to understand than Relational
- Expresses relationships clearly
- Rules to convert ER-diagrams to Relational Schema
- Some systems use ER-model for schema design
- Some people use ER-model as step before creating relational tables