Relational Databases

CPSC 315 – Programming Studio

Project 1, Lecture 2

Slides adapted from those used by Jeffrey Ullman, via Jennifer Welch

Relational Data Model

- Relations are stored in tables
  - e.g. Sponsor(Senator,Bill)

Sponsor

<table>
<thead>
<tr>
<th>Senator</th>
<th>Bill</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smith</td>
<td>Tax</td>
</tr>
<tr>
<td>Jones</td>
<td>Defense</td>
</tr>
<tr>
<td>Smith</td>
<td>Defense</td>
</tr>
<tr>
<td>Adams</td>
<td>Commercio</td>
</tr>
</tbody>
</table>

Attributes

Tuples

Schemas

- A **relation schema** is a relation name and a list of attributes
  - Sponsor(Senator,Bill)
- A **database** is a collection of relations
- A **database schema** is the set of all the relation schemas in the database

Converting from Entity-Relationship Model

- ER: Entity set -> relation
  - ER Attributes become Relational attributes
- ER: Relationship -> relation
  - Keys of connected ER entity sets become Relational attributes
ER Entity Sets

Senator
- Name
- Party
- State
- Years

Bill
- Name
- Text

Lobbyist
- Name
- Organization

ER Relationships

- Senator(Name, Party, State, Years)
- Bill(Name, Text)
- Lobbyist(Name, Organization)

Relations

- Sponsored(Senator, Bill)
- Wrote(Bill, Lobbyist)
- Contributed(Senator, Lobbyist)

Remember, each of these is expressed as a table (with the columns given by the “parameters”)

Notice that columns can refer to “bigger” items, with even more attributes
Combining Relations

• Relations can sometimes be combined.
• Assume a “base” entity set with its relation.
• If there is a many-to-one relation, that can be combined with the base entity set.
• Should not combine many-to-many
  − Redundancy – each of the many stored

Example (many-to-one): (Good)
- Person(Name, Birthdate, Height, Weight, Eye Color, Hair Color)
- BornIn(Person, Town)
- Person(Name, Birthdate, Height, Weight, Eye Color, Hair Color, Town)

Example (many-to-many): (Bad)
- Senator(Name, Party, State, Years)
- Sponsored(Senator, Bill)
- Senator(Name, Party, State, Years, Bill)

Weak Entity Sets

• The relation for a weak entity set must contain all the elements of its key
• Supporting relationships are usually redundant (unless possibly multi-way)

Weak Entity Set Example

Baseball Player
- First Name
- Last Name
- Number
- Position
- Nationality
- Salary
- Birthdate

Note arrow: indicates many to one.
Weak Entity Set Example

- Team(Name, City)
- Baseball Player(Number, TeamName, First Name, Last Name, Position, Birthdate, Nationality, Salary)

Note that we don't need PlaysOn(BaseballPlayer.Number, BaseballPlayer.TeamName, Team.Name)

Redundant (same)
Weak Entity Set Example

- Team(Name, City)
- Baseball Player(Number, TeamName, FirstName, LastName, Position, Birthdate, Nationality, Salary)
- Note that we don’t need PlaysOn(BaseballPlayer.Number, BaseballPlayer.Team.Name) already included

Subclasses
Different Options

- Different ways to represent subclasses

Object-Oriented Style

- One relation for each subset, including all “inherited” attributes

<table>
<thead>
<tr>
<th>Elected Official</th>
<th>U.S. Representative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Party</td>
</tr>
<tr>
<td>Chet Edwards</td>
<td>Democrat</td>
</tr>
<tr>
<td>John Cornyn</td>
<td>Republican</td>
</tr>
<tr>
<td>John Adams</td>
<td>Federalist</td>
</tr>
<tr>
<td>Ron Paul</td>
<td>Republican</td>
</tr>
</tbody>
</table>

Entity-Relationship Style

- One relation for each subclass (including key)

<table>
<thead>
<tr>
<th>Elected Official</th>
<th>U.S. Representative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Party</td>
</tr>
<tr>
<td>Chet Edwards</td>
<td>Democrat</td>
</tr>
<tr>
<td>John Cornyn</td>
<td>Republican</td>
</tr>
<tr>
<td>John Adams</td>
<td>Federalist</td>
</tr>
<tr>
<td>Ron Paul</td>
<td>Republican</td>
</tr>
</tbody>
</table>
Using Nulls Style

- One relation total, with nulls for unknown information
- Can save space, but problematic if multiple subclasses or lots of NULLs

<table>
<thead>
<tr>
<th>U.S. Representative</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Party</td>
<td>District</td>
</tr>
<tr>
<td>Chet Edwards</td>
<td>Democrat</td>
<td>17</td>
</tr>
<tr>
<td>John Cornyn</td>
<td>Republican</td>
<td>NULL</td>
</tr>
<tr>
<td>John Adams</td>
<td>Federalist</td>
<td>NULL</td>
</tr>
<tr>
<td>Ron Paul</td>
<td>Republican</td>
<td>14</td>
</tr>
</tbody>
</table>

Keys

- A Key “functionally determines” all other attributes of the relation
  - Given a relation and a key, there is only one tuple that corresponds to it
- There are subtle differences from an E-R key, which we won’t go into.