ER-to-Relational Mapping

Lecture 8
Outline

1. Context
2. The Algorithm
Database Design and Implementation Process

- **Phase 1:** Requirements collection and analysis
  - Data requirements
  - Processing requirements

- **Phase 2:** Conceptual database design
  - Conceptual Schema design (DBMS-independent)
  - Transaction and application design (DBMS-independent)

- **Phase 3:** Choice of DBMS
  - Logical Schema and view design (DBMS-dependent)
  - Frequencies, performance constraints

- **Phase 4:** Data model mapping (logical design)
  - Internal Schema design (DBMS-dependent)
  - DDL statements
  - Transaction and application implementation
  - SDL statements

- **Phase 5:** Physical design

- **Phase 6:** System implementation and tuning

**ER-to-Relational Mapping**

March 1, 2018
Data Models

External Level
- External View
- External/Conceptual Mapping

Conceptual Level
- Conceptual/Internal Mapping
- Conceptual Schema

Internal Level
- Internal Schema

End Users
- . . .
- External View

Stored Database
Example ERD
Resulting Relational Schema

**EMPLOYEE**
- Fname
- Minit
- Lname
- Ssn
- Bdate
- Address
- Sex
- Salary
- Super_ssn
- Dno

**DEPARTMENT**
- Dname
- Dnumber
- Mgr_ssn
- Mgr_start_date

**DEPT_LOCATIONS**
- Dnumber
- Dlocation

**PROJECT**
- Pname
- Pnumber
- Plocation
- Dnum

**WORKS_ON**
- Essn
- Pno
- Hours

**DEPENDENT**
- Essn
- Dependent_name
- Sex
- Bdate
- Relationship
Step 1: Regular Entity Types

i. For each regular/strong entity type, create a corresponding relation that includes all the simple attributes (includes simple attributes of composite relations)

ii. Choose one of the key attributes as primary
   - If composite, the simple attributes together form the primary key

iii. Any remaining key attributes are kept as secondary unique keys (these will be useful for physical tuning w.r.t. indexing analysis)
Example ERD
## Step 1 Result

<table>
<thead>
<tr>
<th>EMPLOYEE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fname</td>
</tr>
<tr>
<td>--------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DEPARTMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dname</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PROJECT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pname</td>
</tr>
<tr>
<td>-------</td>
</tr>
</tbody>
</table>

---

**ER-to-Relational Mapping**

March 1, 2018
Step 2: Weak Entity Types

i. For each weak entity type, create a corresponding relation that includes all the simple attributes

ii. Add as a foreign key all of the primary key attribute(s) in the entity corresponding to the owner entity type

iii. The primary key is the combination of all the primary key attributes from the owner and the partial key of the weak entity, if any
Step 2 Result

ER-to-Relational Mapping

March 1, 2018
Step 3: Mapping Binary 1-to-1

Three approaches

– **Foreign Key**
  • Usually appropriate

– Merged Relation
  • Possible when both participations are total

– Relationship Relation
  • Not discussed
Step 3: Mapping Binary 1-to-1

*Foreign Key*

i. Choose one relation as $S$, the other $T$
   - Better if $S$ has total participation (reduces number of NULL values)

ii. Add to $S$ all the simple attributes of the relationship

iii. Add as a foreign key in $S$ the primary key attributes of $T$
## Step 2 Result

### EMPLOYEE

<table>
<thead>
<tr>
<th>Fname</th>
<th>Minit</th>
<th>Lname</th>
<th>Ssn</th>
<th>Bdate</th>
<th>Address</th>
<th>Sex</th>
<th>Salary</th>
</tr>
</thead>
</table>

### DEPARTMENT

<table>
<thead>
<tr>
<th>Dname</th>
<th>Dnumber</th>
</tr>
</thead>
</table>
### Step 3 Result

**EMPLOYEE**

<table>
<thead>
<tr>
<th>Fname</th>
<th>Minit</th>
<th>Lname</th>
<th>Ssn</th>
<th>Bdate</th>
<th>Address</th>
<th>Sex</th>
<th>Salary</th>
</tr>
</thead>
</table>

**DEPARTMENT**

<table>
<thead>
<tr>
<th>Dname</th>
<th>Dnumber</th>
<th>Mgr_ssn</th>
<th>Mgr_start_date</th>
</tr>
</thead>
</table>

This table represents the result of the ER-to-Relational Mapping process detailed in the context of the course CS3200 – Database Design, taught by Derbinsky in Spring 2018.
Step 4: Binary 1-to-N

i. Choose the S relation as the type at the N-side of the relationship, other is T

ii. Add as a foreign key to S all of the primary key attribute(s) of T

Another approach: create a relationship relation
Example ERD
Step 4 Result

ER-to-Relational Mapping

March 1, 2018
Step 5: Binary M-to-N

i. Create a new relation S (termed: relationship relation)
   – In some ERD dialects, actually drawn in

ii. Add as foreign keys the primary keys of both relations; their combination forms the primary key of S

iii. Add any simple attributes of the M:N relationship to S
Example ERD

ER-to-Relational Mapping

March 1, 2018
Step 5 Result

ER-to-Relational Mapping

March 1, 2018
Step 6: Multivalued Attributes

i. Create a new relation $S$

ii. Add as foreign keys the primary keys of the corresponding relation

iii. Add the attribute to $S$ (if composite, the simple attributes); the combination of all attributes in $S$ forms the primary key
Example ERD
Step 6 Result

ER-to-Relational Mapping

March 1, 2018
Step 7: Specialization/Generalization

A. Multiple relations – subclass and superclass
   - Usually works (assumes unique id at parent)

B. Multiple relations – subclass only
   - Should only be used for disjoint

C. Single relation with one type attribute
   - Only for disjoint, can result in many NULLs

D. Single relation with multiple type attributes
   - Better for overlapping, could be disjoint
Specialization/Generalization (A)
Specialization/Generalization (B)

<table>
<thead>
<tr>
<th>CAR</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle_id</td>
<td>License_plate_no</td>
<td>Price</td>
<td>Max_speed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TRUCK</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle_id</td>
<td>License_plate_no</td>
<td>Price</td>
<td>No_of_axles</td>
</tr>
</tbody>
</table>

ER-to-Relational Mapping

March 1, 2018
Specialization/Generalization (C)
Specialization/Generalization (D)

<table>
<thead>
<tr>
<th>PART</th>
<th>Part_no</th>
<th>Description</th>
<th>Mflag</th>
<th>Drawing_no</th>
<th>Manufacture_date</th>
<th>Batch_no</th>
<th>Pflag</th>
<th>Supplier_name</th>
<th>List_price</th>
</tr>
</thead>
</table>

ER-to-Relational Mapping

March 1, 2018
Summary

• Mapping from ERDs to relations is an algorithmic process

• Some choice points involve comparing time-space tradeoffs (more in physical design)