**IS 441 Week 5 Class Summary and Highlights**

Dr. Yue Zhang

Class outline:

|  |  |
| --- | --- |
| 1. Pick up HW-related slides; 2. Chap 3 – EER (Enhanced E-R) | 1. Exercise on converting ERD 🡪 relational model 2. Chap 4, Part 2: Functional Dependency |

1. Pick up HW2-related Chap 4 Slides:
2. 1-1 relationship converted to relational model:
   * + - 1. In 1-1 relationship there is usually one entity that is mandatory, and another that is optional;
         2. When converting to relational model, the one that is mandatory has the behavior of a “1-side”, and the one that is optional has the behavior of a “M-side”;
         3. Therefore, the PK of the entity on the mandatory side should appear in the optional side’s relation (table) as the foreign key.

Book: NURSE (mandatory) and CARE-STATION (optional)

1. Unary relationship converted to relational model – Unary 1-M

On E-R diagram we have an entity relate to itself; which

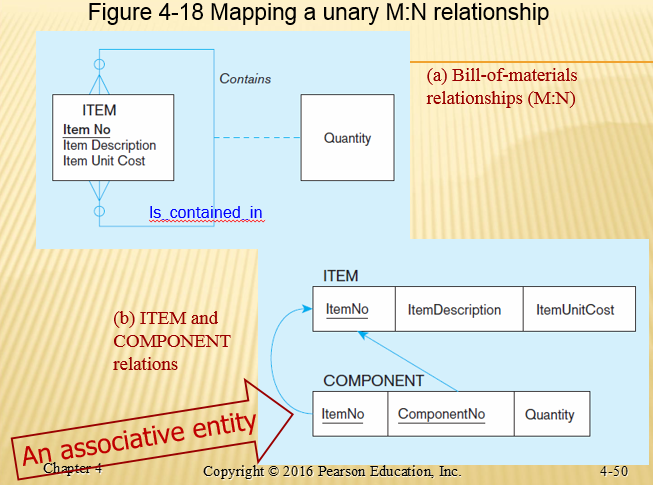
* + - * 1. Does NOT mean that in the converted relation (table) a row is related to itself;
        2. DOES mean that a row is related to ANOTHER row in THE SAME TABLE.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Manages  EMPLOYEE  EmpID  LName  FName  MgrID | Some employees Manages (many other employees); some employees are Managed by ONE manager.   |  |  |  |  | | --- | --- | --- | --- | | EmpID | LName | FName | MgrID | | E001 | Adams |  | E003 | | E003 | Brown |  |  | | E004 | Cohen |  |  | | E007 | Davis |  | E004 | | E009 | Evans |  | E004 |   Adams’ manager is Brown; Davis’ and Evans’ manager is Cohen.  Some rows’ MgrID is the EmpID of his/her manager. |

1. Unary relationship converted to relational model – Unary M-M

|  |  |  |
| --- | --- | --- |
| **Two Entities (Binary)** | **“Folded”** | **One Entity (Unary)** |
| A  Ka  Fa1  B  Kb  Fb1 | Folded | U  Ku  Fu1 |
| C  Ka. Kb  Fc1, Fc2  A  Ka  Fa1  B  Kb  Fb1 | Folded | U  Ku  Fu1  C  Ku1. Ku2  Fc1, Fc2 |
| A  Fa1  Ka  C  Fc2  Fc1  Kb  Ka  Kb  Fb1  B | Folded  A and B folded into one (U) | U  Fu1  Ku  C  Fc2  Fc1  Ku2  Ku1 |

Example from text:



1. Lecture on Chapter 3 - EER:
2. Generalization and specialization

|  |  |
| --- | --- |
|  |  |

1. Constraints in Supertype/subtype relationships –
   * + - 1. COMPLETENESS:
2. Total: Each instance of the supertype must participate in one (or more) of the subtypes.

Symbol: Double-lines (“the double lines force the instances down to a subtype”)

1. Partial: Instances of the supertype do NOT have to participate in any subtype.

Symbol: Single-line (“the Single-line does not force the instances down to a subtype”)

* + - * 1. DISJOINTNESS:

1. Disjoint: Each instance of the supertype can be in ONLY one of the subtypes (“d”)

Example: Vehicles specialize to CAR or SUV; a vehicle can ONLY be in ONE subtype.

1. Overlap: Each instance of the supertype can be in more than one of the subtypes (“o”)

Example: STUDENT specialized into PhD, Master, Undergrad; a student can be in more than one category (a PhD student is pursuing a master’s in another major, etc).

1. Practice of completeness and disjointness: Slide 3-23 (Figure 3-1)
   * + - 1. Go through the EER top-down; at each relationship, determine its constraints (total vs partial, disjoint vs overlap), as ask “why” (support your judgment with the relationship line (single or double), and disjointness constraint (“d” or “o”).
2. Exercise on Converting ERD to Relational Model
3. Strategy of the conversion (using Problem 4-46 on P. 199 as example):

How many entities are there? (6; so we anticipate six relations [tables]);

How many relationships each entity participates? – so we anticipate the number of referential integrity arrows (PK-FK pairs) entering/leaving each relation (table);

TEAM participates in two relationships; so we expect two referential integrity arrows entering or leaving TEAM;

PARTICIPATION participates in three relationships; ; so we expect three referential integrity arrows entering or leaving PARTICIPATION;

Etc.

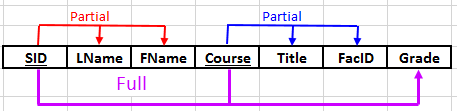
Examine the cardinality for each entity: among the relationships this entity participates, in which relationship is this entity on 1-side, and in which other this entity is on M-side; this will determine the direction of the referential integrity arrows: in-arrow for the entity on 1-side, and out-arrow for the entity on the M-side.

1. Plan the layout of the graphical representation of the relations:

Those who are directly related on EER should be placed closer;

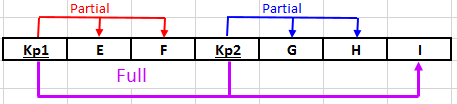
Try to place associative entities between the two (three) regular entities this associative entity is connecting, to avoid excessive cross-over of referential integrity arrows.

1. Draw the graphical representation according to the plan developed in the above “1”:
2. Drawing the graphical representation of relational model:
3. Draw all the relations first;
4. Then add the referential integrity arrows one by one: finishing on entity/relation, then move on to the next entity/relation.
5. Check the correctness of the relational model based on
6. Number of referential integrity arrows (i.e., # of relationships) each relation has (participates), which corresponds to Planning step ii;
7. Direction of referential integrity arrows to/from a relation, which corresponds to Planning iii.
8. Chap 4, Part 2: “Functional Dependency” in Normalization
9. Functional Dependency: The value of one attribute (the **determin*ant***) determines the value of another attribute (the determin***ee***)
   1. Example: SID 🡺 LName, FName, Gender, DOB, Major, Address…
   2. 
10. Second Normal Form: There is no partial dependency.
    1. In the case where there is a composite key, the key has 2 or more parts.
       1. A non-key field must fully depend on this key (consists of multiple fields) – we call this type of functional dependency “Full dependency”. Full dependency is desirable.
    2. When a non-key field depends only on a PART of the key (rather than the complete key), there is a partial dependency. Partial dependency is NOT desirable.
    3. Example of partial and full dependencies:



* + 1. In the above figure, the composite key have two fields SID and Course.
    2. The non-key fields LName and FName depend on SID, which is only A PART of the composite key;
    3. The non-key fields Title and FacID depend on COURSE, which is only A PART of the composite key;
    4. The non-key field Grade depends on the full composite key – SID + Course, so that dependency is a Full Dependency.
    5. Partial dependencies can cause anomalies which will be discussed next class.

1. More general presentation of partial dependency:



* 1. Kp1 – Key Part 1; Kp2 = Key part 2; the composite key = Kp1 + Kp2;
  2. E and F depend on Kp1, which is partial dependency;
  3. G and H depend on Kp2, which is partial dependency;
  4. I depends on BOTH Kp1 and Kp2, or the composite key, so the relationship

(Kp1 + Kp2) 🡺 I (“🡺” for “determines”)

Is a Full Dependency.