Fundamentals/ICY: Databases 2013/14
Week 6: Monday

John Barnden
Professor of Artificial Intelligence

School of Computer Science
University of Birmingham, UK

Module Eval Forms Today
Hooray!
You have to wait till the end!

Reminder of Friday

Weak *Entity Types*

◆ A *weak* <u>entity type</u> **E** is one such that there is a relationship *R* from some other entity type **F** to **E** that satisfies the following two conditions:

- *R* is <u>**strong**</u>.

- **E** is *<u>existence-dependent</u>* on **F** via *R*.

  That is, an E entity *e* can only exist in the database if some F entity in the database bears relationship *R* to *e*.

  This actually just adds the requirement that *R* $-_1$ (the *R* relationship but in the E to F direction) be *mandatory*.

Weak Entity Types, contd.

◆ So on a previous slide, Dependants is *weak*, because there is a **strong** relationship to it from Customers, and Dependants is **existence-dependent** on Customers via this relationship.

- **Mary's** existence *in the database as a member of Dependants* relies on the existence of **customer 1698674** *in the database*.

  But this **doesn't mean** Mary would vanish from the world if customer 1698674 left the database or even if that customer were to vanish from the world.

  And indeed Mary could herself be an entity in type Customers, and could remain so even if 1698674 left the DB or really vanished from the world.

New

Strong *Entity Types*

◆ A ***strong*** entity type is one that is not weak! ….

◆ So, in particular, any entity type that receives only <u>weak</u> relationships from other entity types is <u>strong</u>.

  • ***BUT this is NOT the ONLY way for an entity type to be strong.***

  [make sure you understand why]

◆ So the usual case is for an entity type to be strong.

Mental Exercises for You

◆ What about the Employments bridging type we introduced?

◆ Although it might look at first as though a strong relationship necessarily implies existence dependence, it doesn't. Why not?

Entity Relationship (ER) Model(s) and Diagrams

(ERMs and ERDs)

The Entity Relationship Model

◆ Introduced by Chen in 1976

◆ Most widely used "conceptual model" of DBs.

◆ **"The ER model"** : general meaning = just the idea of thinking of things as composed of entities, attributes and relationships.

◆ We also say that applying this approach in a particular case gives rise to **an "ER model"** of the specific environment of interest.

◆ <u>Diagrams based on</u> the/a model are a widely accepted and adopted graphical approach to database design.

◆ The/an ER model has nothing intrinsically to do with diagrams!!! Let alone any specific sort of diagram!!!

A Conceptual Model

◆ ER model that provides high-level, manager/user-friendly view of the database

◆ Basis for identification and description of main data objects and relationships, avoiding details

Entity Relationship Diagrams (ERDs)

◆ An ER model of an environment forms the basis of an ER diagram (ERD) or several ERDs.

◆ There are several markedly different styles of ERD, and for each main style there are several variants.

- **And the style in the module handouts will differ somewhat from the one in the textbook and the two in these lectures.**

- **That's four styles overall, in this module !!**

◆ An ERD depicts (some of) the ER model's entity types, attributes and relationships, and (depending on the diagram style) varying amounts of other info such as connectivities, cardinalities, keys, weakness, …

◆ You are encouraged to think of variants and mixtures of ERD styles. You can use these in your work as long as you EXPLAIN what changes you've made. Each "official" style has its own advantages and disadvantages (unclarities, omissions, …
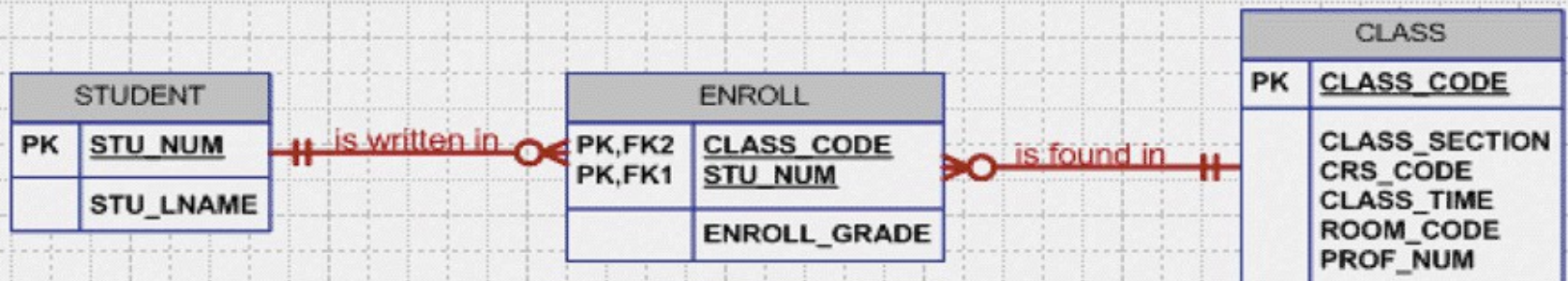
Quick Flavour of Two Styles of Diagram


FIGURE 4.26 A COMPOSITE ENTITY IN AN ERD
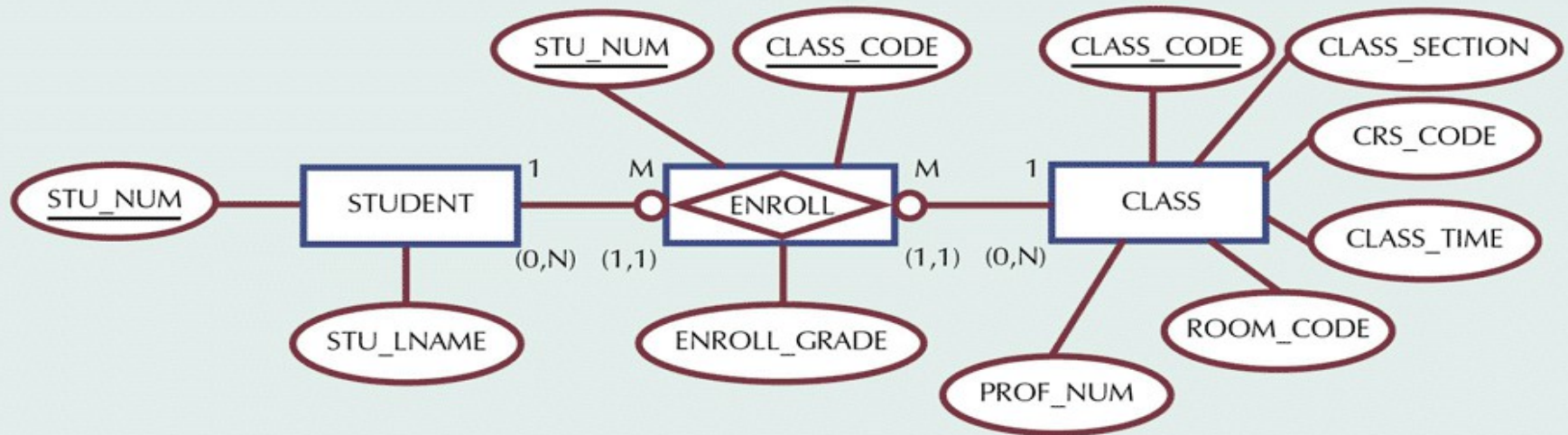
Caution about some Crow's Ft Diagrams

- In Crow's Foot, a DASHED relationship line is for a relationship that is WEAK BOTH WAYS ROUND (the normal case).

- A solid line is used if the relationship is strong at least ONE way round.

- In some cases in diagrams on following slides (from previous textbook editions), a solid line is used but a dashed one would be more plausible.

- (The Chen notation does not distinguish between strong and weak relationships.)

The Completed Tiny College ERD

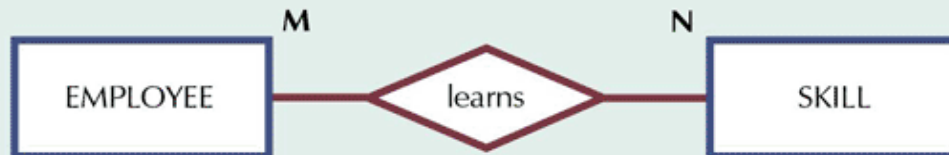FIGURE 4.47 THE COMPLETED TINY COLLEGE ERD
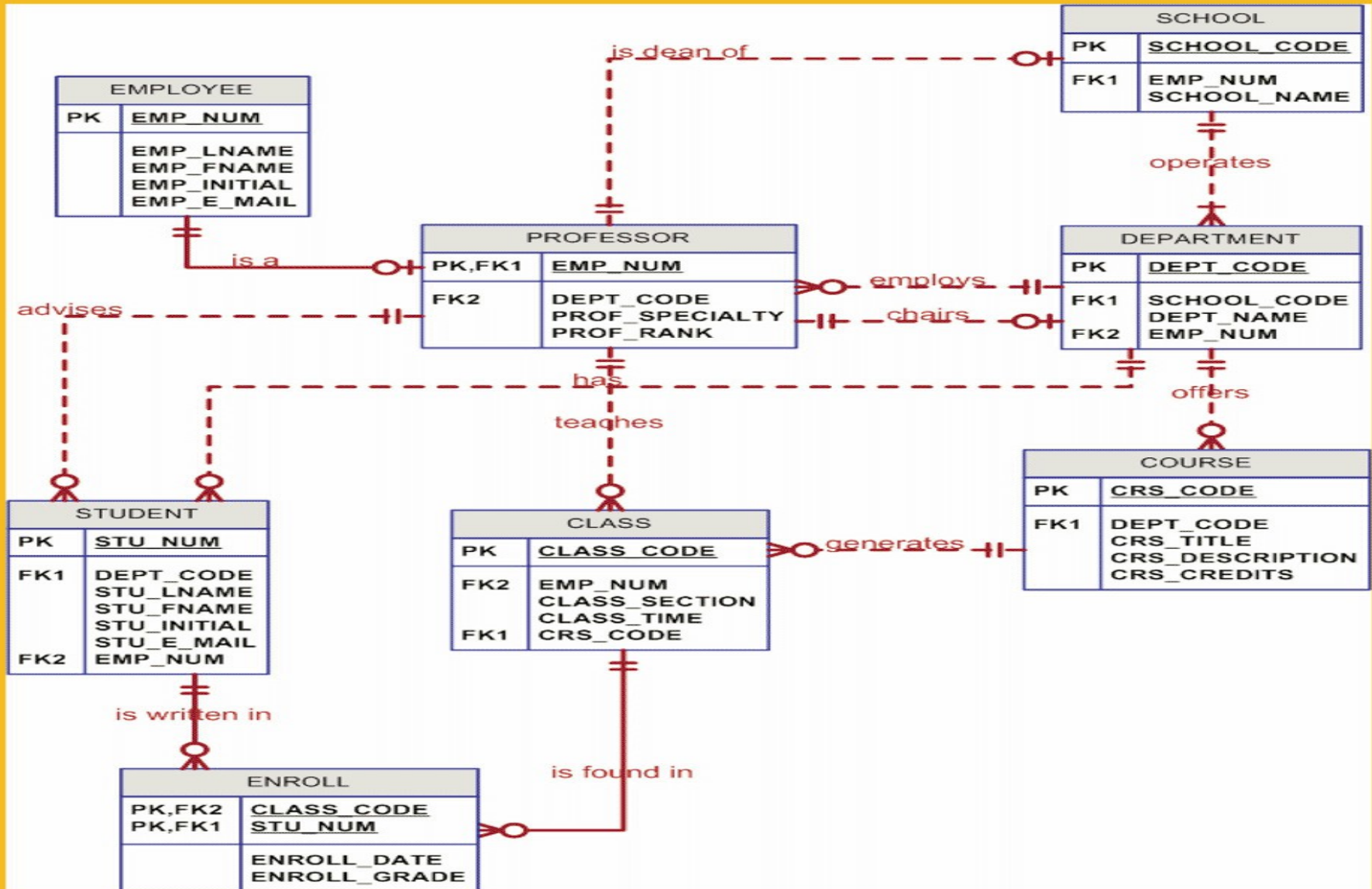
Relationships: The Basic Chen ERD-let

FIGURE 2.6 RELATIONSHIPS: THE BASIC CHEN ERD

A One-to-Many (1:M) Relationship: a PAINTER can paint many PAINTINGs; each PAINTING is painted by one PAINTER

A Many-to-Many (M:N) Relationship: an EMPLOYEE can learn many SKILLs; each SKILL can be learned by many EMPLOYEEs

A One-to-One (1:1) Relationship: an EMPLOYEE manages one STORE; each STORE is managed by one EMPLOYEE

The Basic Crow's Foot ERD-let
(lines should probably be dashed)

FIGURE 2.7   RELATIONSHIPS: THE BASIC CROW'S FOOT ERD

A One-to-Many (1:M) Relationship: a PAINTER can paint many PAINTINGs;
each PAINTING is painted by one PAINTER

PAINTER — paints — PAINTING

A Many-to-Many (M:N) Relationship: an EMPLOYEE can learn many SKILLs;
each SKILL can be learned by many EMPLOYEEs

EMPLOYEE — learns — SKILL

A One-to-One (1:1) Relationship: an EMPLOYEE manages one STORE;
each STORE is managed by one EMPLOYEE

EMPLOYEE — manages — STORE

Caution

◆ In previous two diagrams, each relationship was *mandatory* in both directions.

◆ But saying "1:1", "1:M" or "M:N" does not *of itself* imply mandatoriness in either direction. In particular, don't be deceived by the "1" *here* – it's not a minimum.

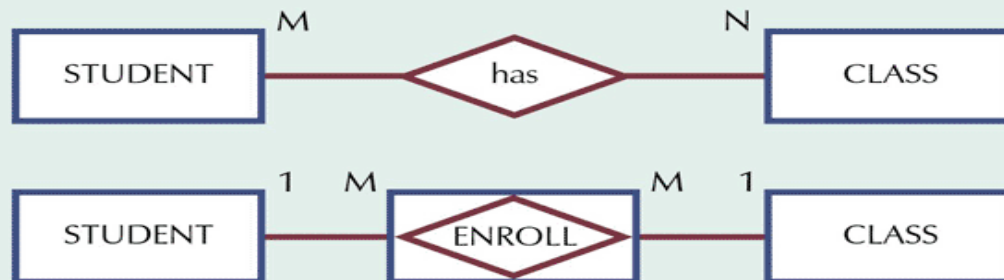◆ We will see in a minute how to draw optional (non-mandatory) relationships.

Conversion of M:N to Two 1:M Relationships
(towards "logical" level)

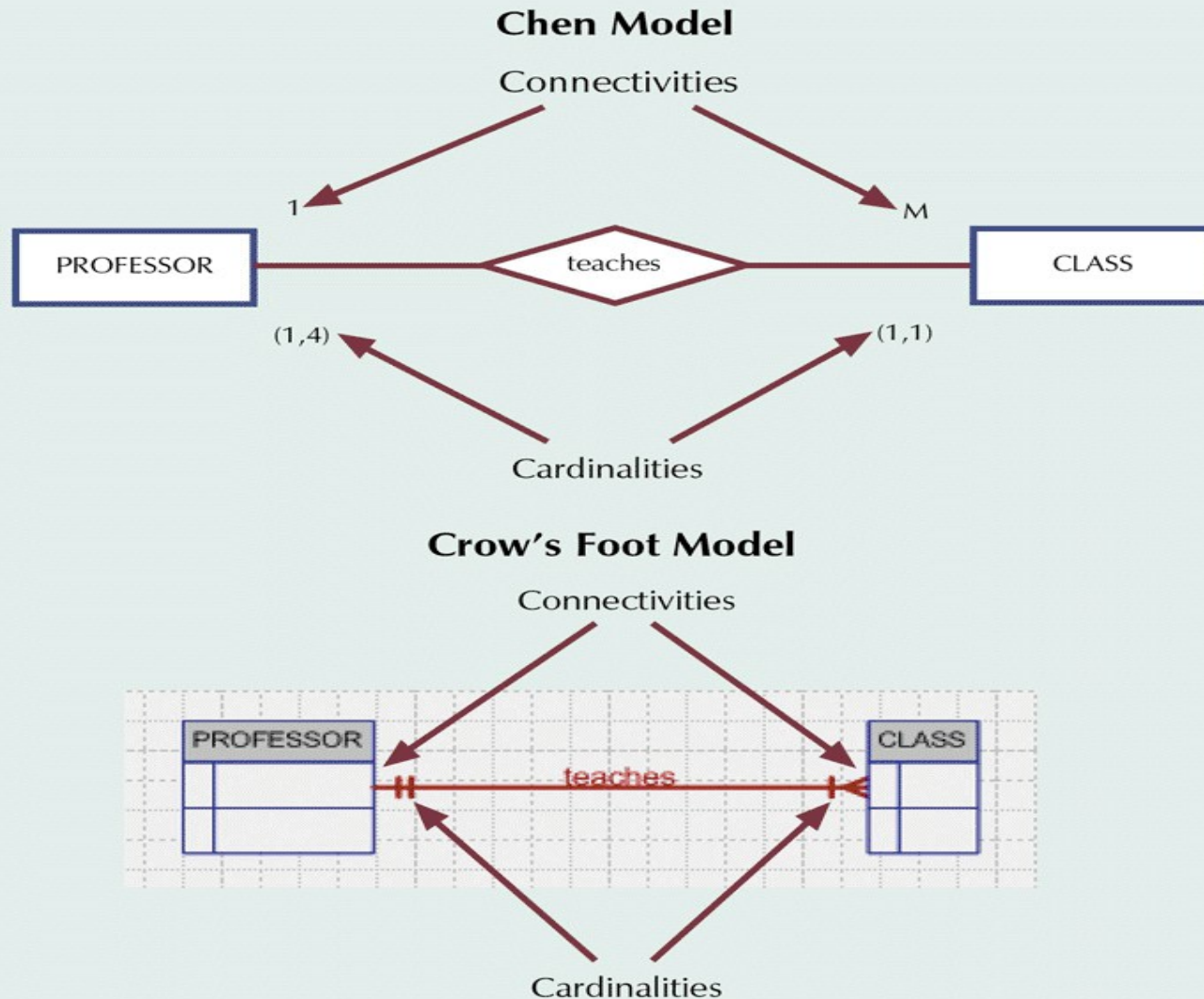FIGURE 3.27 CHANGING THE M:N RELATIONSHIP TO TWO 1:M RELATIONSHIPS
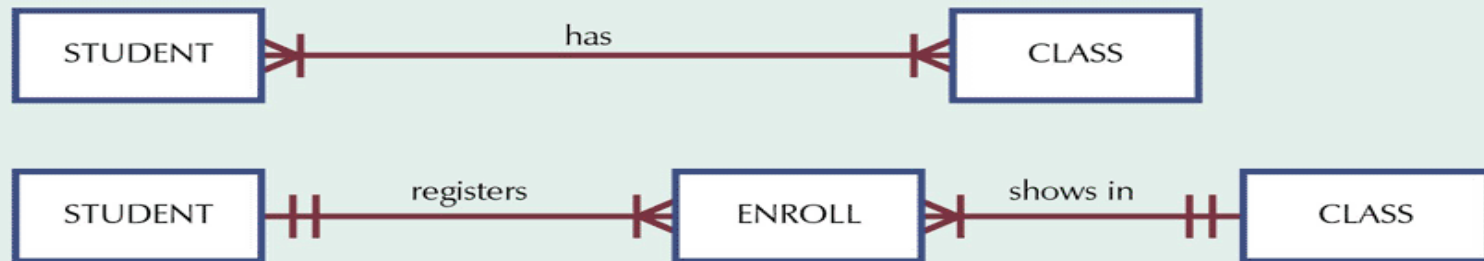
Ex: Which Crow's Ft lines should be dashed, if any??
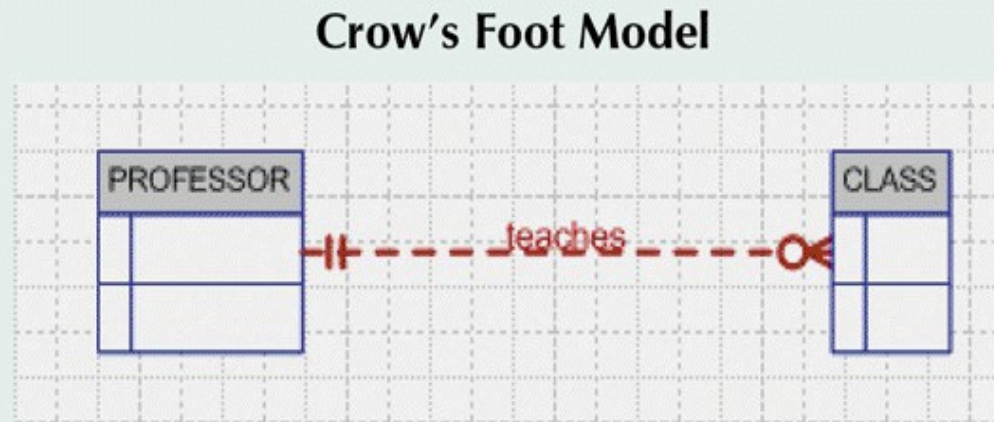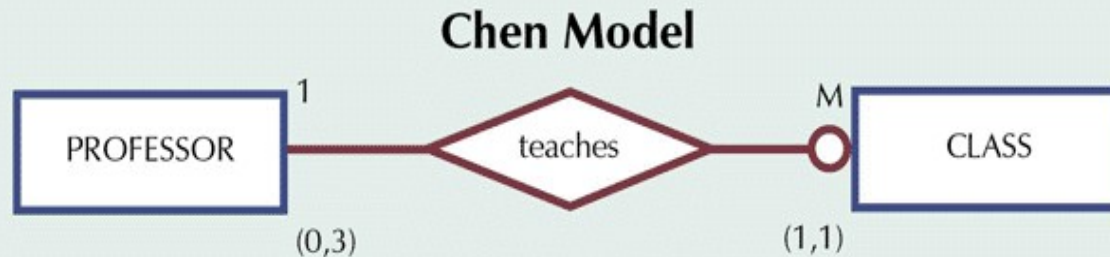
Connectivity and Cardinality in an ERD

FIGURE 4.7 CONNECTIVITY AND CARDINALITY IN AN ERD

Drawing Optionality

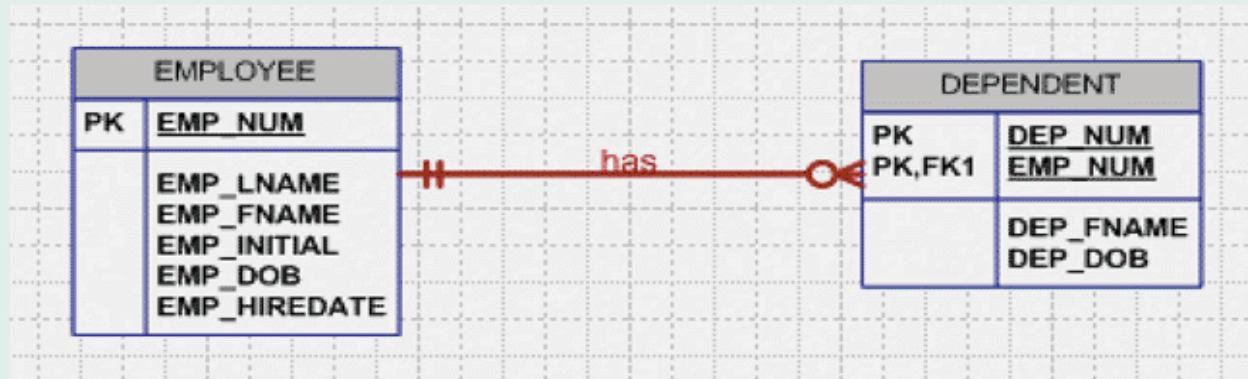FIGURE 4.11   AN OPTIONAL CLASS ENTITY IN THE RELATIONSHIP PROFESSOR TEACHES CLASS

Chen Model

PROFESSOR $\overset{1}{\rule{0pt}{0pt}}$ ── teaches ──$\overset{M}{\rule{0pt}{0pt}}$○ CLASS

(0,3)   (1,1)

Crow's Foot Model

PROFESSOR ─┤├─ — — teaches — — ─○< CLASS

NOTE: the dashing of the line is NOT because of the optionality, but because of weakness

A Weak Entity in an ERD

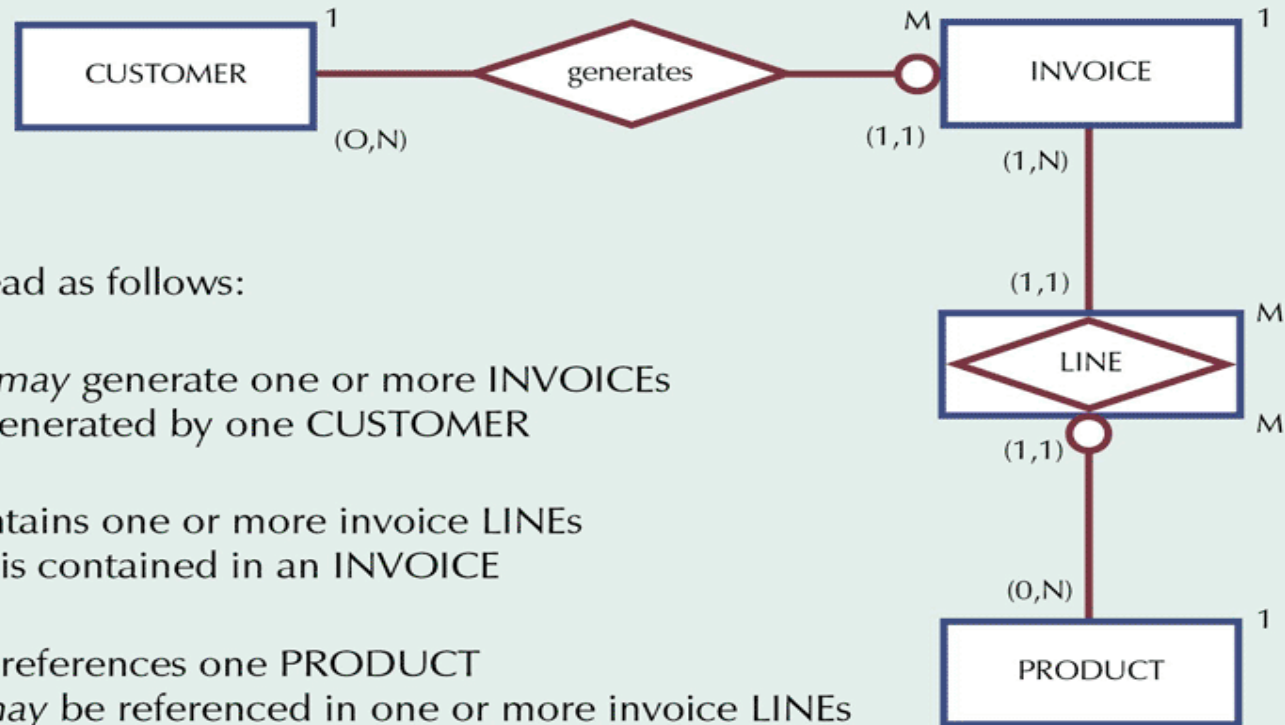

FIGURE 4.14 A WEAK ENTITY IN AN ERD
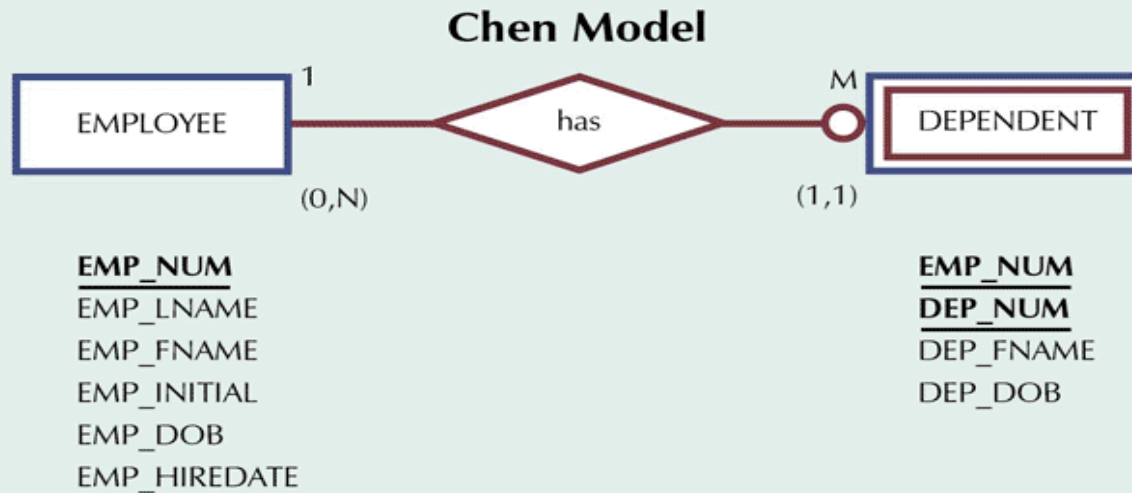
The Chen Representation of the Invoicing Problem

FIGURE 4.32 THE CHEN REPRESENTATION OF THE INVOICING PROBLEM

CUSTOMER — 1 — generates — M — (1,1) — INVOICE — 1

(O,N)

(1,N)

(1,1)

LINE — M

M — (1,1)

(O,N)

PRODUCT — 1

This model may be read as follows:

each CUSTOMER *may* generate one or more INVOICEs
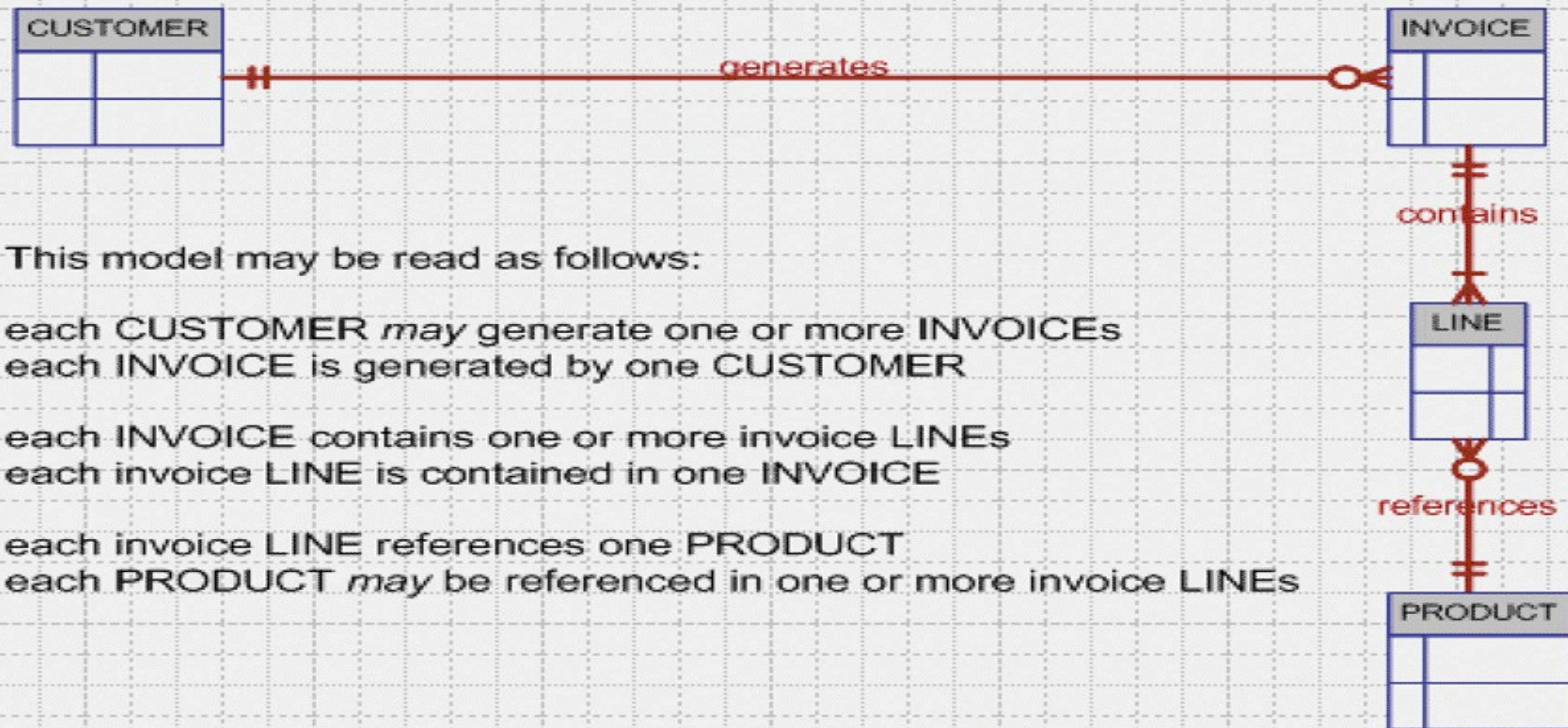each INVOICE is generated by one CUSTOMER

each INVOICE contains one or more invoice LINEs
each invoice LINE is contained in an INVOICE

each invoice LINE references one PRODUCT
each PRODUCT *may* be referenced in one or more invoice LINEs

Bridging entity types are weak, but this is not normally shown

The Crow's Foot Representation of the Invoicing Problem

FIGURE 4.33 THE CROW'S FOOT REPRESENTATION OF THE INVOICING PROBLEM

CUSTOMER —— generates —— INVOICE

This model may be read as follows:

each CUSTOMER *may* generate one or more INVOICEs
each INVOICE is generated by one CUSTOMER

each INVOICE contains one or more invoice LINEs
each invoice LINE is contained in one INVOICE

each invoice LINE references one PRODUCT
each PRODUCT *may* be referenced in one or more invoice LINEs

INVOICE —— contains —— LINE —— references —— PRODUCT

Ex: Which lines should be dashed, if any??