



บทที่ 5 Entity-Relationship (ER) Model

วิจักขณ์ ศรีสัจจะเลิศวาจา

ภาควิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่

เอกสารอ้างอิง

รศ.เพชรรัตน์ ไชติกอภา, ระบบฐานข้อมูล, คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่, 2551.



5.1 ขั้นตอนการออกแบบฐานข้อมูล

- การออกแบบฐานข้อมูล แบ่งเป็น 3 ระดับ ดังนี้
 1. Conceptual database design
 2. Logical database design
 3. Physical database design



5.1 ขั้นตอนการออกแบบฐานข้อมูล

1. Conceptual database design

- สร้าง data model ที่แสดง abstract structure
- พิจารณาจากการดำเนินงานขององค์กรและสภาพแวดล้อมที่เกี่ยวข้อง
- ผู้ออกแบบต้องเข้าใจปัญหาเป็นอย่างดี วิเคราะห์ข้อมูลที่เกี่ยวข้องและตรวจสอบให้ตรงกับความต้องการของผู้ใช้



5.1 ขั้นตอนการออกแบบฐานข้อมูล

2. Logical database design

- แปลง data model จากระดับที่ 1 ให้เป็น data model ที่จะนำไปใช้งานได้ (Data mapping)
- ต้องกำหนดว่าจะใช้ database model แบบใด
- ถ้าจะใช้ฐานข้อมูลเชิงสัมพันธ์ ก็ต้องเลือกตัวแบบฐานข้อมูลเชิงสัมพันธ์
 - โดยแปลง data model ในข้อ 1 ให้เป็นรีเลชัน
- ในขั้นนี้ต้องตรวจสอบความถูกต้องของตัวแบบ และความซ้ำซ้อนของข้อมูล ที่จะทำให้เกิด data anomalies
 - กรณีตัวแบบฐานข้อมูลเชิงสัมพันธ์ ใช้วิธี Normalization



5.1 ขั้นตอนการออกแบบฐานข้อมูล

3. Physical database design

- จัดทำเมื่อมีการเลือก DBMS เรียบร้อย
- เกี่ยวข้องกับการเลือก data storage, file organization, data access, ชนิดอุปกรณ์, การทำ index และอื่น ๆ
- ขั้นตอนนี้จะมีผลต่อประสิทธิภาพการทำงานจริง
- วัดจาก Response time และ Throughput
- การทดสอบขั้นนี้ต้องคำนึงถึงฮาร์ดแวร์และเครือข่ายสื่อสารอื่น ๆ ที่เกี่ยวข้อง

5.2 หลักการเบื้องต้นของ ER model

- ER model ถูกนำเสนอโดย Peter Chen เมื่อ ค.ศ. 1976
- โดยเสนอเป็น Entity Relationship Diagram หรือ ERD
- มีผู้เสนอเพิ่มอีกหลายคน เช่น Theorey, Yang และ Fray
- การเขียนสัญลักษณ์ (Notation) มีหลายรูปแบบ
 - Chen notation เป็นรูปแบบดั้งเดิม
 - Crow's foot notation
 - UML notation (Universal Modeling Language)
- ในที่นี่จะใช้รูปแบบของ Chen ซึ่งไม่ขึ้นกับตัวแบบฐานข้อมูลใด
- ER model เป็นตัวแบบในระดับหลักการ ผลลัพธ์ที่ได้จะนำไปแปลงเป็นตัวแบบฐานข้อมูล Logical ที่ต้องการ



5.2.1 Entity and Entity set

- ส่วนประกอบหลักของ ERD คือ เอนทิตี หมายถึง สิ่งที่ต้องการเก็บบันทึกข้อมูล และเป็นสิ่งที่เกิดอย่างเป็นอิสระในโลกแห่งความจริง อาจจะเป็น
 - สิ่งที่สามารถจับต้องได้ เช่น บุคคล สถานที่ รถยนต์
 - สิ่งที่ไม่สามารถจับต้องได้ เช่น การนัดหมาย หลักสูตร เหตุการณ์ กิจกรรม



5.2.1 Entity and Entity set

- การกำหนดเอนทิตีที่เกี่ยวข้องกับระบบ เริ่มจากระบุเป้าหมายว่าต้องการเก็บบันทึกข้อมูลอะไร เช่น
- ต้องการสร้างฐานข้อมูลนักศึกษาคณะวิทยาศาสตร์
 - กำหนดเป้าหมายเบื้องต้นว่าต้องการเก็บข้อมูลอะไร
 - จะได้ว่า นักศึกษาแต่ละคน แทน 1 เอนทิตี
 - ถ้ามีนศ. 1000 คน ก็จะมี 1000 เอนทิตี
 - หากต้องการเก็บข้อมูลกระบวนวิชาด้วย
 - ข้อมูลแต่ละกระบวนวิชา ก็แทน 1 เอนทิตี
 - มี 300 กระบวนวิชา ก็จะมี 300 เอนทิตี
 - เอนทิตีที่เกิดขึ้นแต่ละตัวเรียกว่า Entity instance หรือ Entity occurrence เช่น นศ. รหัส 4805901
 - โดยที่แต่ละเอนทิตีจะต้องไม่ซ้ำกัน สามารถ Identify ได้

5.2.1 Entity and Entity set

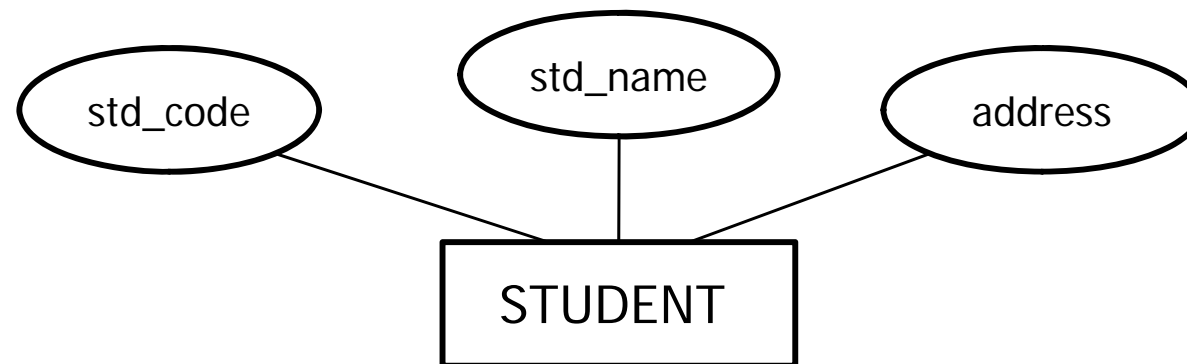
- ในฐานะข้อมูล หรือ ERM จะไม่พูดถึงการเก็บข้อมูลเพียง 1 เอนทิตี
- แต่จะรวมเอนทิตีที่มีคุณสมบัติเดียวกันไว้ เรียกว่า เอนทิตีเซต แทนด้วย

STUDENT

- แนวทางตั้งชื่อเอนทิตี
 - ชื่อเอนทิตีควรจะสื่อความหมายชัดเจน ใช้คำหนึ่งคำ
 - ชื่อเอนทิตีเป็นคำนามเอกพจน์ เช่น CUSTOMER, CAR
 - ชื่อเอนทิตีเป็นชื่อที่ใช้ทั่วไปในองค์กร ผู้ใช้คุ้นเคย หรือเข้าใจได้ และไม่มีการใช้ซ้ำ
 - ควรใช้ชื่อเอนทิตีเดียวกันในทุกผังอาร์ทที่อ้างถึงเอนทิตีเซตนั้น
 - ควรนิยามชื่อเอนทิตีที่ตั้งขึ้นว่าหมายถึงสิ่งใด มีข้อมูลใดเป็น Identifier

5.2.2 Attribute

- หลังจากกำหนดเอนทิตีหลักใน ERM ได้แล้ว
- ขั้นตอนต่อมาต้องหว่า มีข้อมูลใดบ้างที่ต้องการสำหรับแต่ละเอนทิตี เช่น
- STUDENT ต้องการบันทึก รหัสประจำตัว ชื่อ และที่อยู่
- EMPLOYEE ต้องการบันทึก ชื่อ ตำแหน่ง แผนก วันที่บรรจุ และเงินเดือน
- เป็นสิ่งที่บ่งบอกลักษณะหรือคุณสมบัติของเอนทิตีที่เราสนใจ แทนด้วย



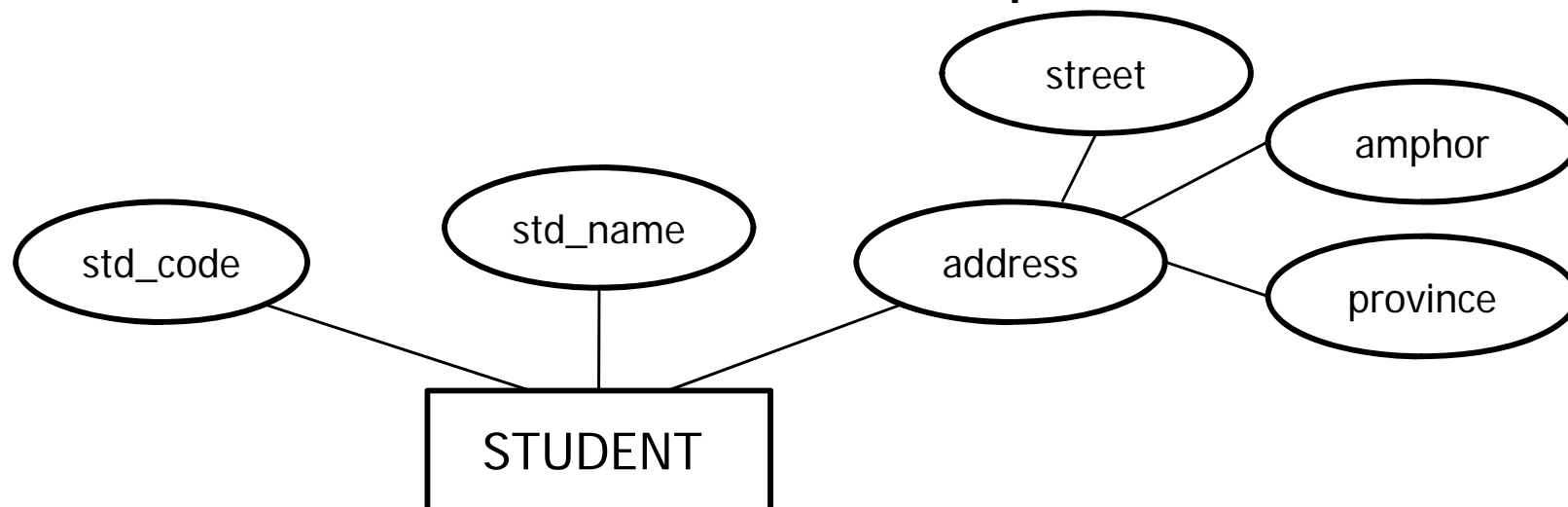


5.2.2 Attribute

- แต่ละแอททริบิวต์จะปรากฏได้เพียงครั้งเดียวในเอนทิตีเซต
- ค่าแอททริบิวต์จะต้องมาจากโดเมนที่กำหนด หรืออาจเป็นค่า Null
- แนวทางการตั้งชื่อแอททริบิวต์
 - ชื่อแอททริบิวต์ควรมีความหมายบ่งชี้ค่าที่บันทึกอยู่ภายใน
 - ใช้คำนาม
 - ชื่อแอททริบิวต์ในเอนทิตีเซตเดียวกันต้องไม่ซ้ำกัน แม้ต่างเอนทิตีเซตก็ไม่ใช้ซ้ำกัน เพื่อป้องกันความสับสน
 - อาจกำหนดรูปแบบมาตรฐานของการตั้งชื่อ
 - ไม่ควรใช้อักขระพิเศษ เช่น ! @ \$ *
 - เป็นคำที่ใช้ทั่วไป และเป็นที่ยอมรับในองค์กร

5.2.2.1 Simple attribute and Composite attribute

- Simple attribute หรือ Atomic attribute คือ
แอททริบิวต์ที่ไม่สามารถแตกค่าเป็นส่วนย่อยได้อีก เช่น รหัสนักศึกษา
- Composite attribute หรือ Group attribute คือ
แอททริบิวต์ที่เกิดจากการรวมแอททริบิวต์ที่เกี่ยวข้องของหลายตัวเข้าด้วยกัน
เช่น แอททริบิวต์ที่อยู่ ประกอบด้วย บ้านเลขที่ อำเภอ จังหวัด
- การเขียน ERM อาจจะแสดง Composite attribute ไว้ชัดเจน



5.2.2.2 Single-valued attribute and Multi-valued attribute

- Single-valued attribute

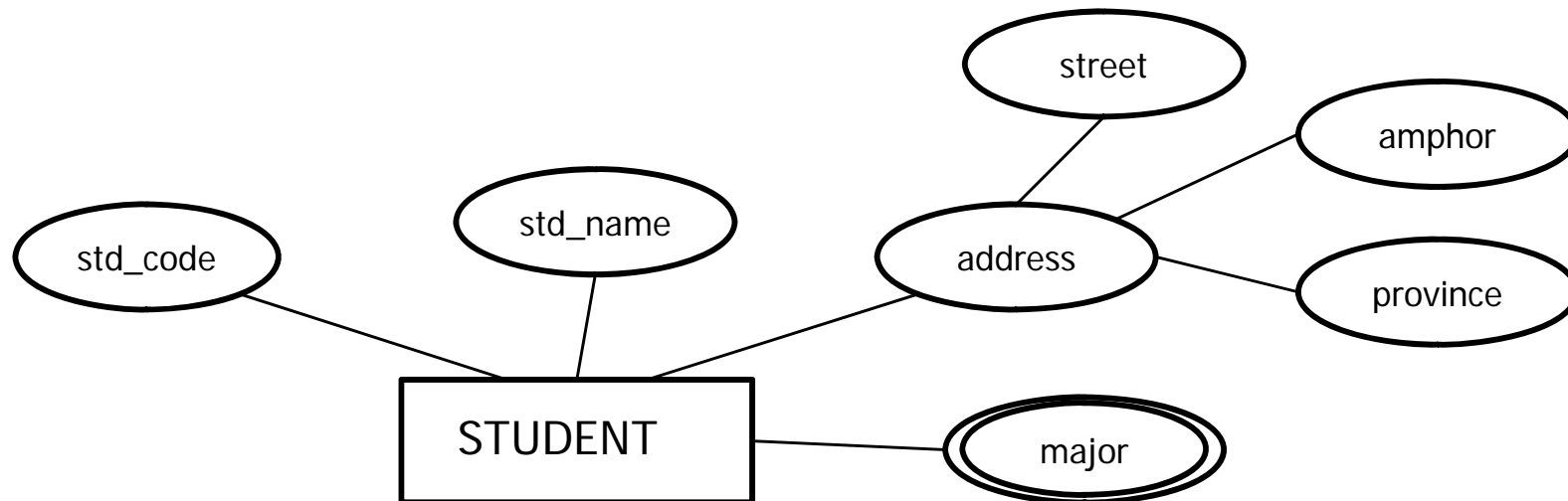
แอททริบิวต์ที่มีค่าเพียงหนึ่งค่าสำหรับแต่ละเอนทิตี เช่น

`std_name` เป็นค่าชื่อของนักศึกษาซึ่งมีค่าเดียว

- Multi-valued attribute

สำหรับแต่ละเอนทิตีในแอททริบิวต์หนึ่งอาจจะมีหลายค่า เช่น

นักศึกษาบางคนมีสาขาวิชาเอกมากกว่าหนึ่งสาขา



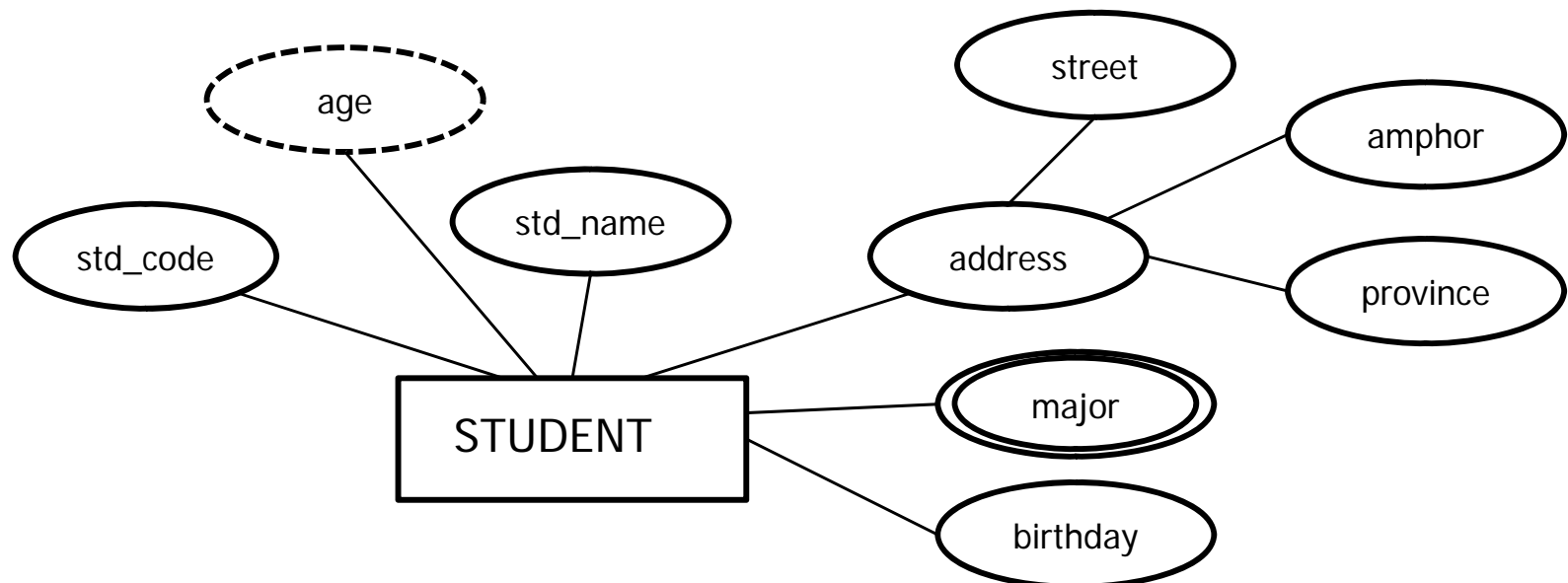
5.2.2.3 Stored attribute and Derived attribute

- Stored attribute

แอททริบิวต์ปกติจะเกิดจากการบันทึกข้อมูลเข้ามาสู่ฐานข้อมูลโดยตรง

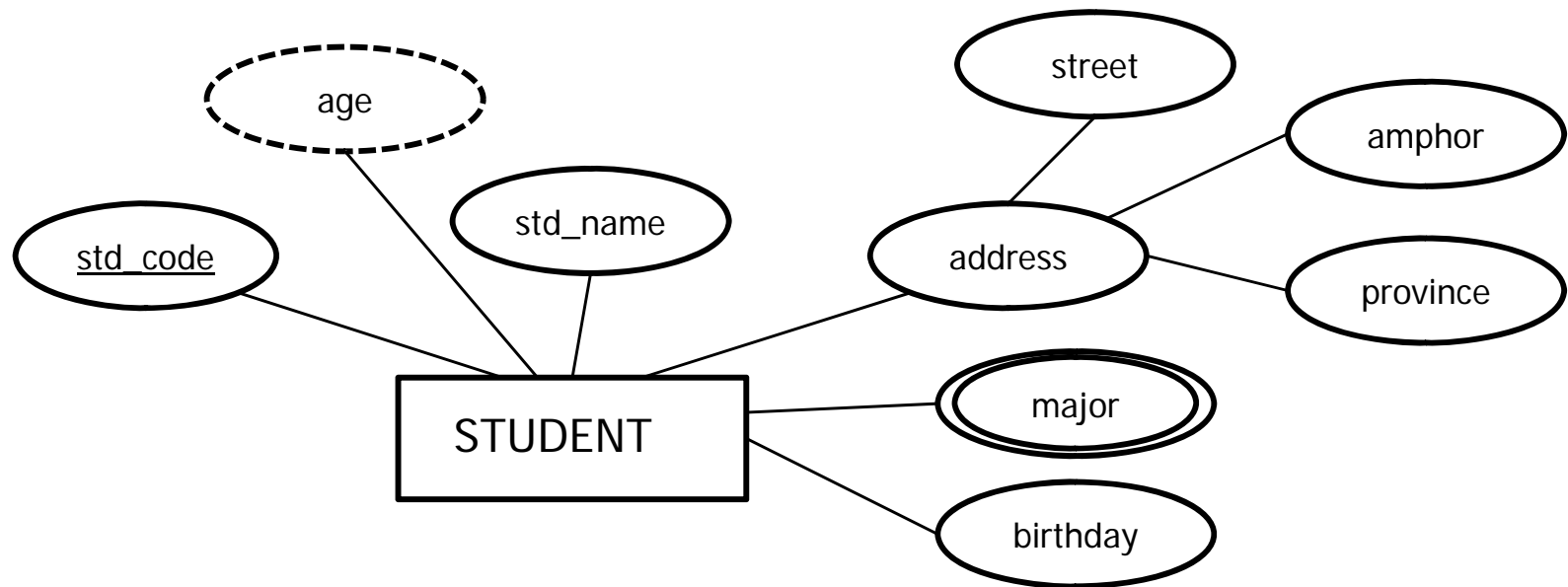
- Derived attribute

แอททริบิวต์ที่เกิดจากการคำนวณค่าจากแอททริบิวต์อื่นที่สัมพันธ์กัน ซึ่งอาจจะเป็นแอททริบิวต์ในรีเลชันเดียวกันหรือไม่ก็ได้ เช่น หากแอททริบิวต์ **age** เกิดจากการคำนวณของแอททริบิวต์ **birthday** เรียก **age** ว่า **Derived attribute**



5.2.2.4 Identifier attribute

- แอททริบิวต์หรือกลุ่มของแอททริบิวต์ ที่สามารถชี้เฉพาะเอนทิตีที่เป็นสมาชิกในเซต
- ในฐานะข้อมูลเชิงสัมพันธ์ เรียกว่า คีย์
- ค่าที่เป็นตัวชี้เฉพาะนี้ จะไม่ซ้ำกันเลย



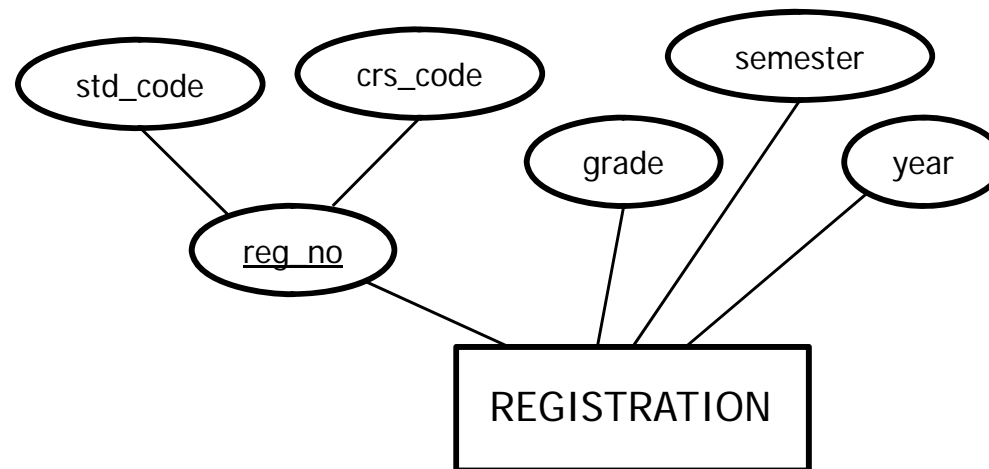


5.2.3 Entity to relation mapping

- ERM เป็นตัวแบบในระดับหลักการ
- เมื่อออกแบบเสร็จจะต้องแปลงเป็นตัวแบบระดับตรรกะ ซึ่งขึ้นอยู่กับทางเลือกชนิดของตัวแบบฐานข้อมูล
- ในที่นี้จะกล่าวถึงตัวแบบฐานข้อมูลเชิงสัมพันธ์

5.2.3 Entity to relation mapping

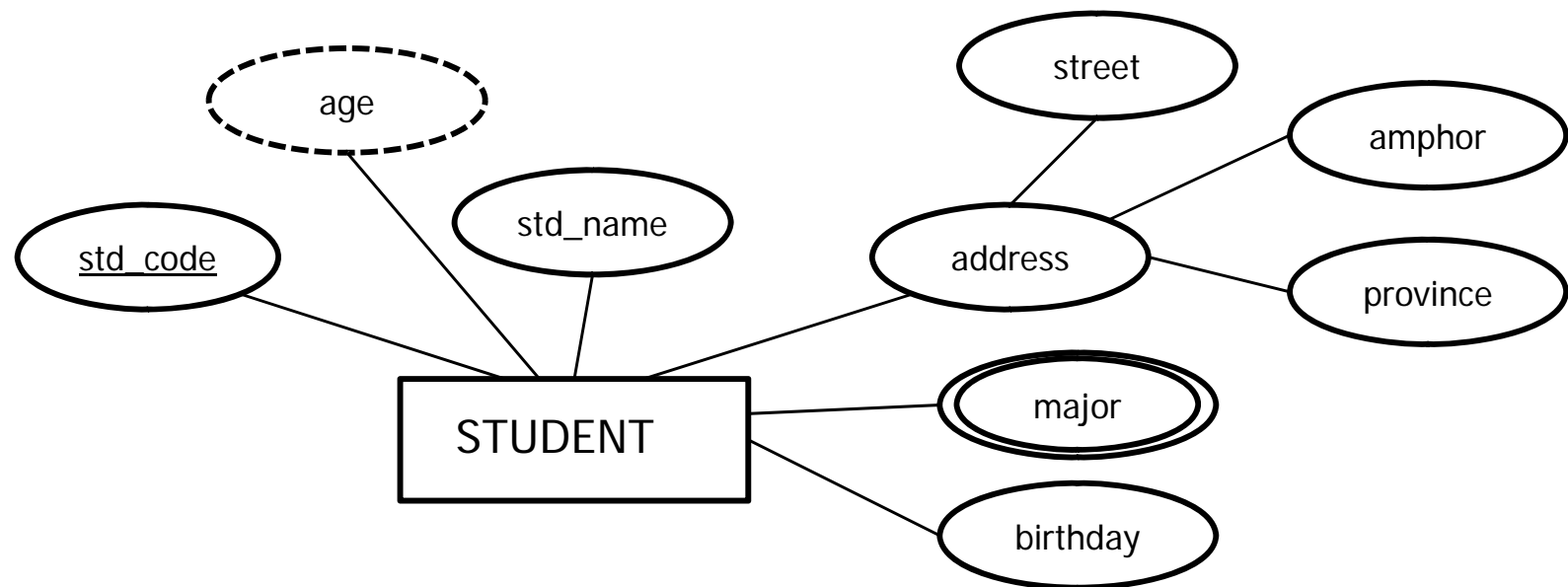
- การแปลงเอนทิตีที่มีแอททริบิวต์แบบต่าง ๆ เป็นรีเลชัน
1. สำหรับแต่ละเอนทิตี ให้สร้างรีเลชันที่ประกอบด้วยแอททริบิวต์เดี่ยวทุกตัว
 - กำหนดแอททริบิวต์ที่เหมาะสมให้เป็นคีย์หลัก
 - ชื่อเอนทิตีเป็นชื่อรีเลชัน
 - แอททริบิวต์ประกอบ ต้องแปลงแอททริบิวต์ที่แบ่งแยกไม่ได้แล้วให้เป็นแอททริบิวต์ของรีเลชัน



REGISTRATION(std_code,crs_code,grade,semester,year)

5.2.3 Entity to relation mapping

- กรณีเอนทิตีประกอบด้วยแอททริบิวต์หลายค่า ให้สร้างรีเลชันใหม่เพิ่ม
 - รีเลชันนี้ประกอบด้วยแอททริบิวต์ที่เฉพาะ และแอททริบิวต์แบบหลายค่านั้น
 - จากนั้นให้กำหนดคีย์หลักเป็นคีย์ประกอบจากแอททริบิวต์ทั้งหมด



STUDENT(std_code,std_name,birthday,age,street,amphor,province)
STD_MAJOR(std_code,major)

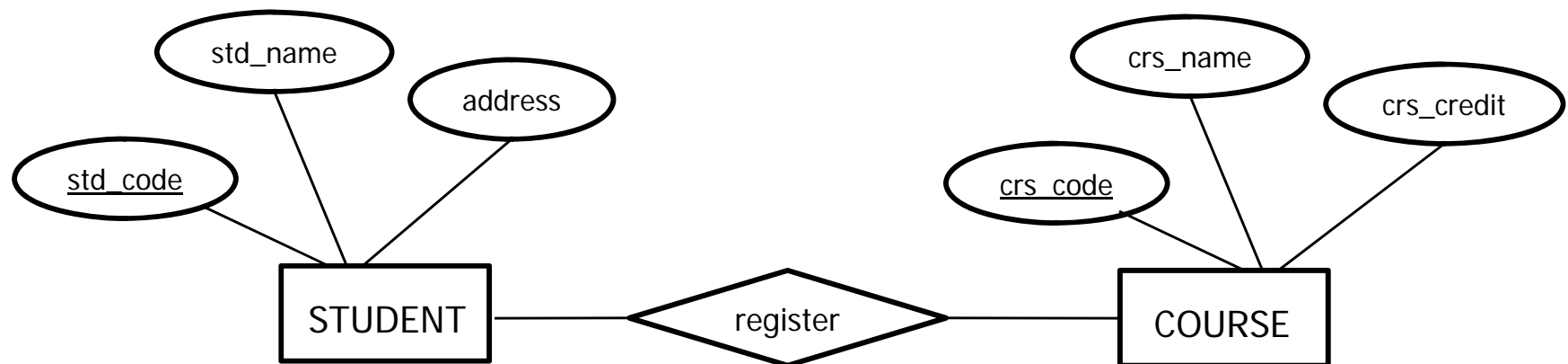
- แอททริบิวต์ **age** ไม่จำเป็นต้องบันทึกในหน่วยบันทึกข้อมูลก็ได้

5.3 Relationship

- ตัวอย่าง ต้องการสร้างระบบฐานข้อมูลสำหรับการลงทะเบียน ซึ่งต้องสามารถติดตามผลลงทะเบียนวิชาต่าง ๆ ของนศ.ทุกคน
- แต่ละเทอมนศ.หนึ่งคนลงทะเบียนเรียนอย่างน้อย 3 วิชา และวิชาที่นักศึกษาลงทะเบียนอาจเป็นวิชาสังกัดสาขาวิชาใดก็ได้
- ระบุเอนทิตีหลักที่เกี่ยวข้อง
- **STUDENT** รหัสประจำตัว ชื่อ สาขาวิชาเอก วิชาที่ลงทะเบียน
- **COURSE** รหัสวิชา ชื่อ จำนวนหน่วยกิต ภาควิชาที่เปิดสอน
- **DEPARTMENT** ชื่อภาควิชา ที่ตั้ง หมายเลขโทรศัพท์
- สังกัดแอททริบิวต์สาขาวิชาเอก และวิชาที่ลงทะเบียนในเอนทิตี **STUDENT**
- สังกัดแอททริบิวต์ภาควิชาที่เปิดสอนในเอนทิตี **COURSE**

5.3.1 Relationship set and Relationship entity

- ความสัมพันธ์ระหว่างเอนทิตีสามารถวิเคราะห์ได้จากกฎธุรกิจ
- **Business rule** เป็นกฎเกณฑ์ที่ทำให้เกิดความสัมพันธ์ระหว่างเอนทิตี เช่น
 - แต่ละเทอมคนคนหนึ่งจะลงทะเบียนอย่างน้อย 3 วิชา
 - กฎนี้ทำให้เกิดความสัมพันธ์ระหว่าง STUDENT กับ COURSE
 - เรียกว่าความสัมพันธ์ ลงทะเบียน (register)



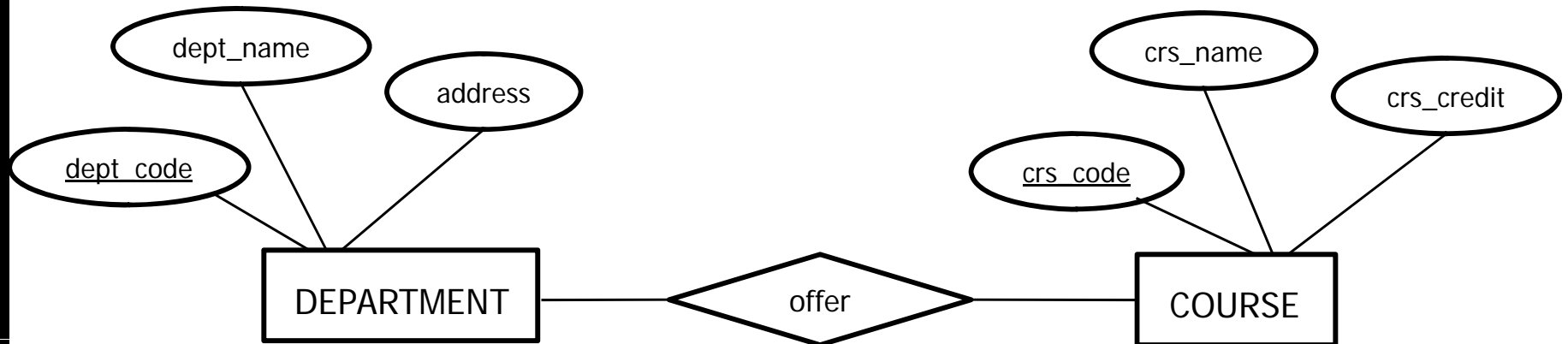
5.3.1 Relationship set and Relationship entity

- Relationship set

ความสัมพันธ์ที่เกิดขึ้นระหว่างเอนทิตีเซต ตั้งชื่อความสัมพันธ์เป็นคำกริยาแบบ present tense

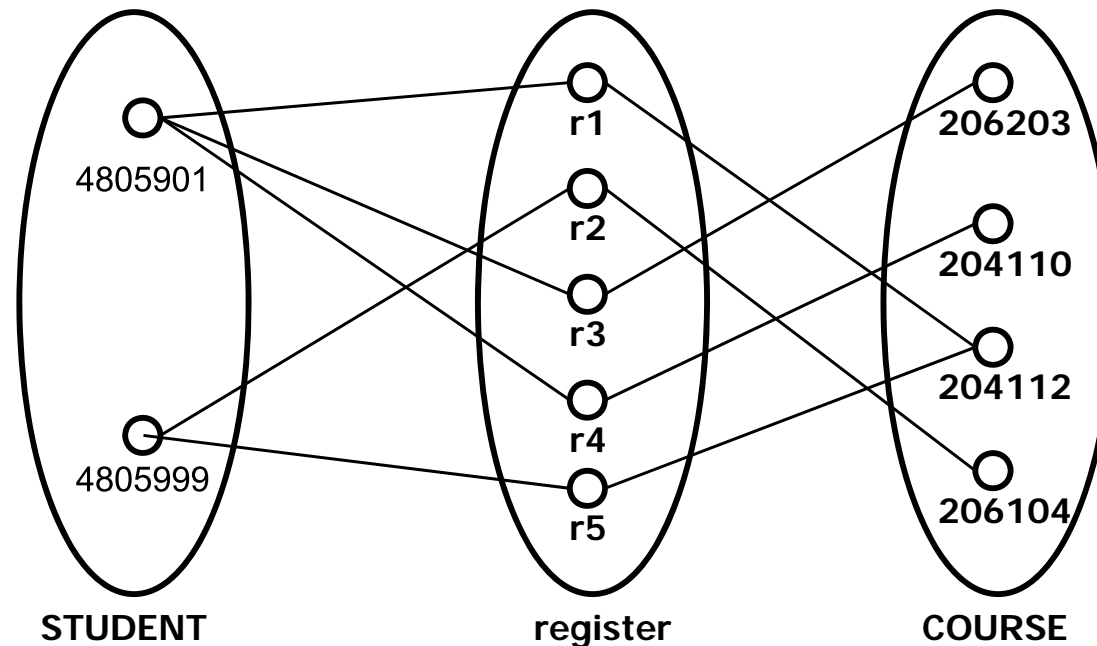
- Relationship entity

ความสัมพันธ์ที่เกิดขึ้นระหว่างเอนทิตีแต่ละตัวในเอนทิตีเซต



5.3.1 Relationship set and Relationship entity

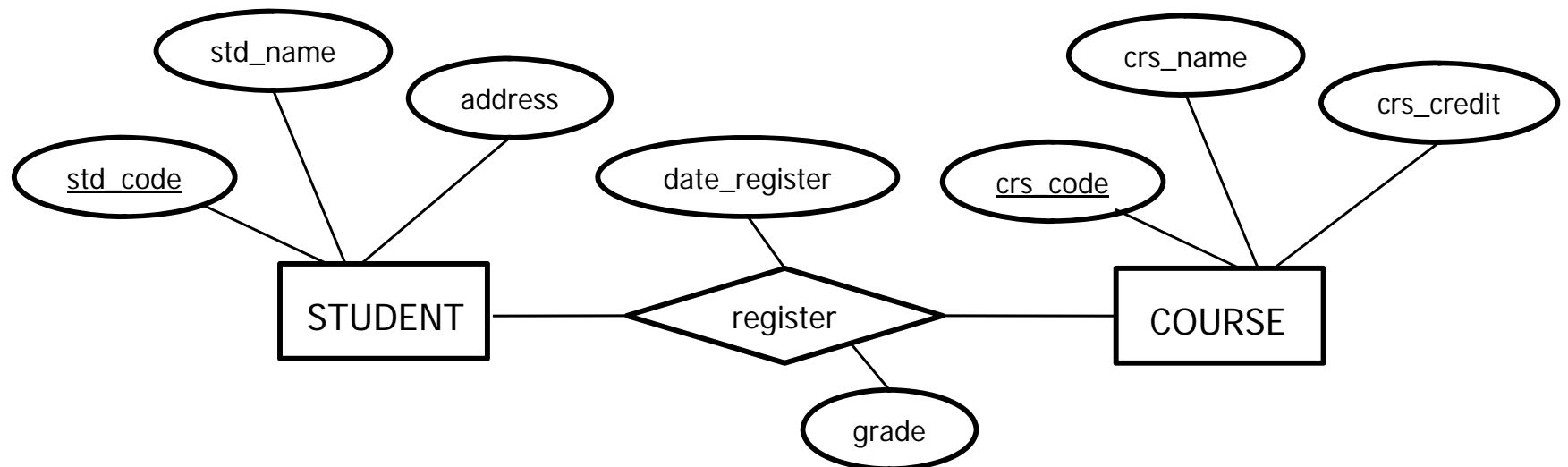
- สมมติความสัมพันธ์ register ดัง occurrence diagram



- เอนทิตีเซต คือ STUDENT และ COURSE
- เอนทิตี คือ แต่ละสมาชิกของเอนทิตีเซต
- Relationship set คือ register
- Relationship entity คือ r1,r2,r3,r4,r5

5.3.2 Attributes of relationship type

- เซตความสัมพันธ์อาจมีแอททริบิวต์เป็นของตัวเอง เช่น
- ถ้าวิทยาลัยบันทึกวันที่ในศ.ลงทะเบียนแต่ละวิชา และเกรดที่ในศ.ได้รับ
- ทั้ง 2 แอททริบิวต์ไม่ใช่คุณสมบัติของ STUDENT และ COURSE
- แต่เป็นคุณสมบัติเฉพาะของการลงทะเบียนแต่ละครั้ง





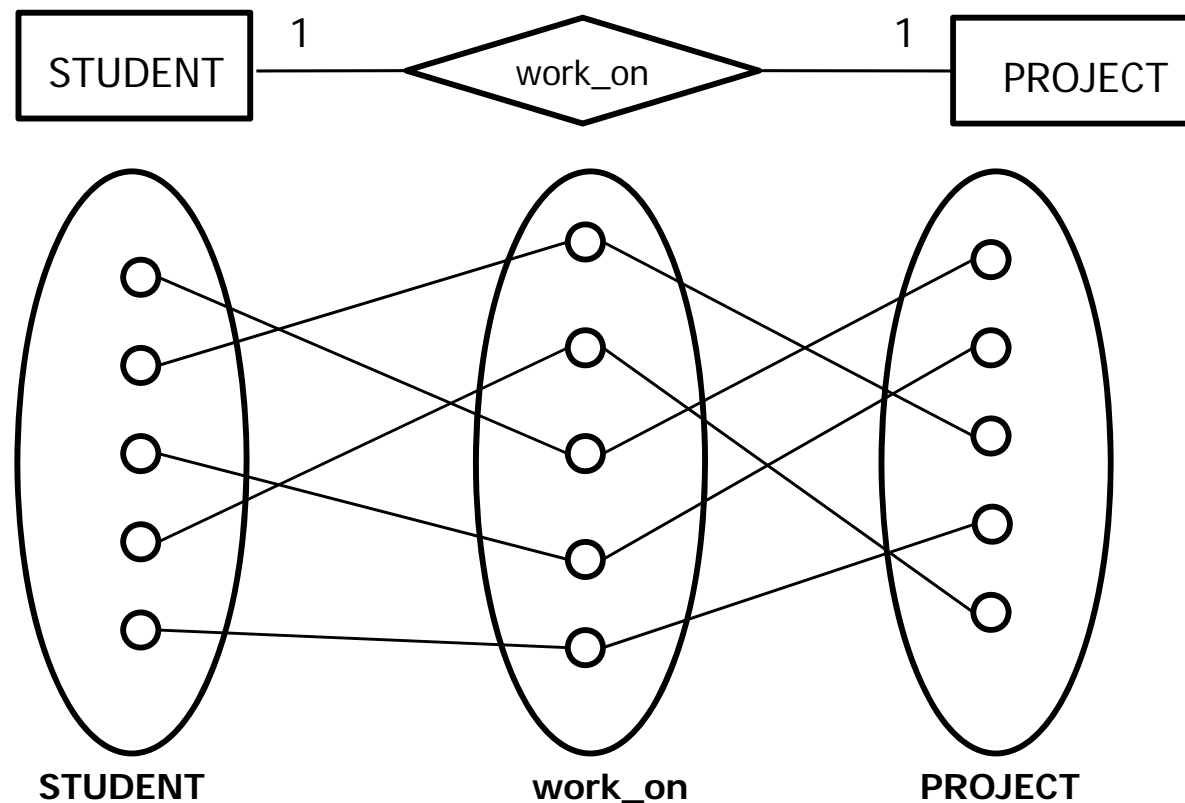
5.3.3 Relationship cardinality

- การระบุความสัมพันธ์ระหว่างเอนทิตีที่มีหลายลักษณะ วิธีหนึ่ง คือ
- การระบุคาร์ดินัลลิตี ระหว่างเอนทิตีเซตหนึ่งกับอีกเซตหนึ่ง
- โดยกำหนดเป็นค่าสูงสุดของคาร์ดินัลลิตี (จำนวนเอนทิตี) ที่เป็นไปได้
- นิยมแทนด้วย **one (1)** หรือ **many (N)** แบ่งเป็น 3 ลักษณะ
 1. One-to-one relationship
 2. One-to-many relationship
 3. Many-to-many relationship

5.3.3 Relationship cardinality

1. One-to-one relationship 1:1

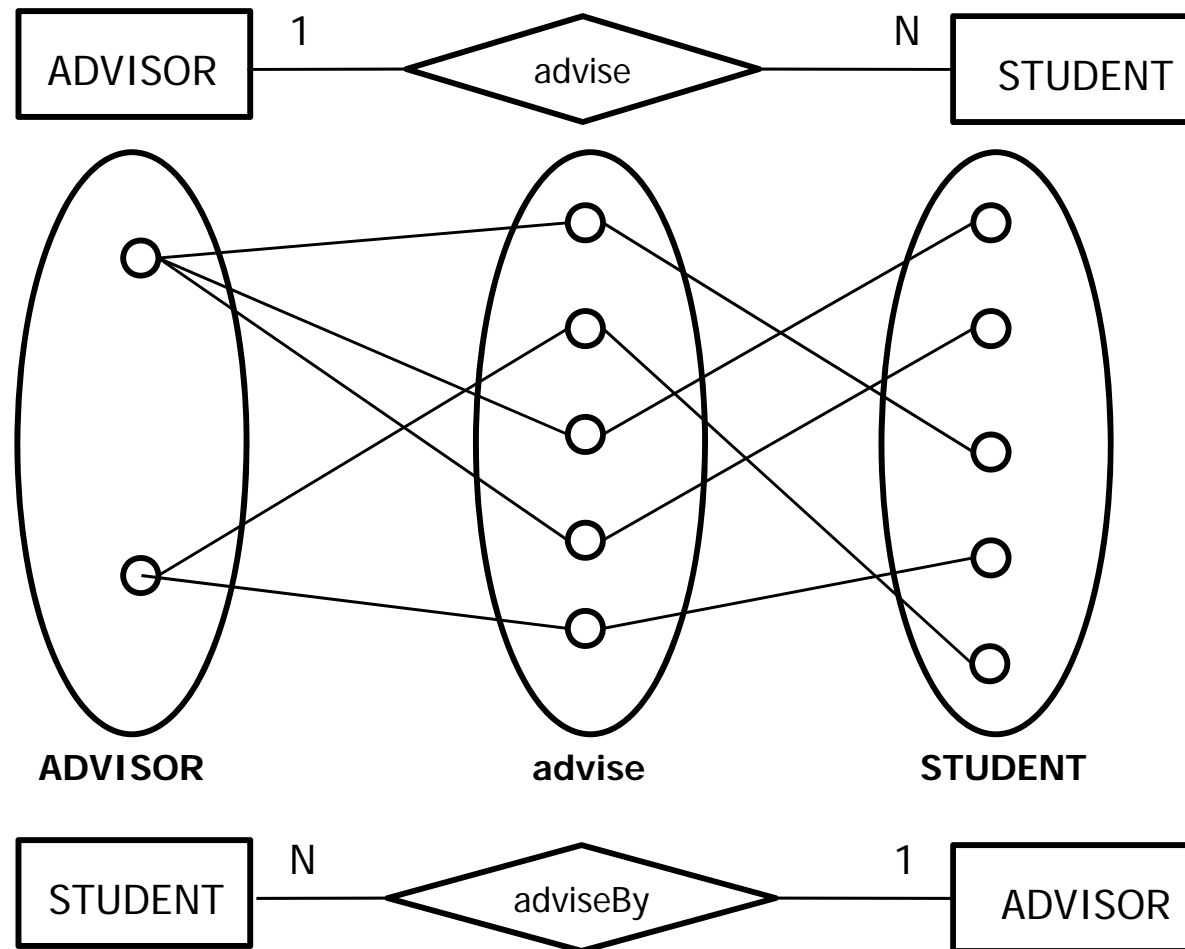
- เอนทิตีในเซตแรกหนึ่งตัวมีความสัมพันธ์กับเซตอื่นเพียงตัวเดียว



5.3.3 Relationship cardinality

2. One-to-many relationship 1:N

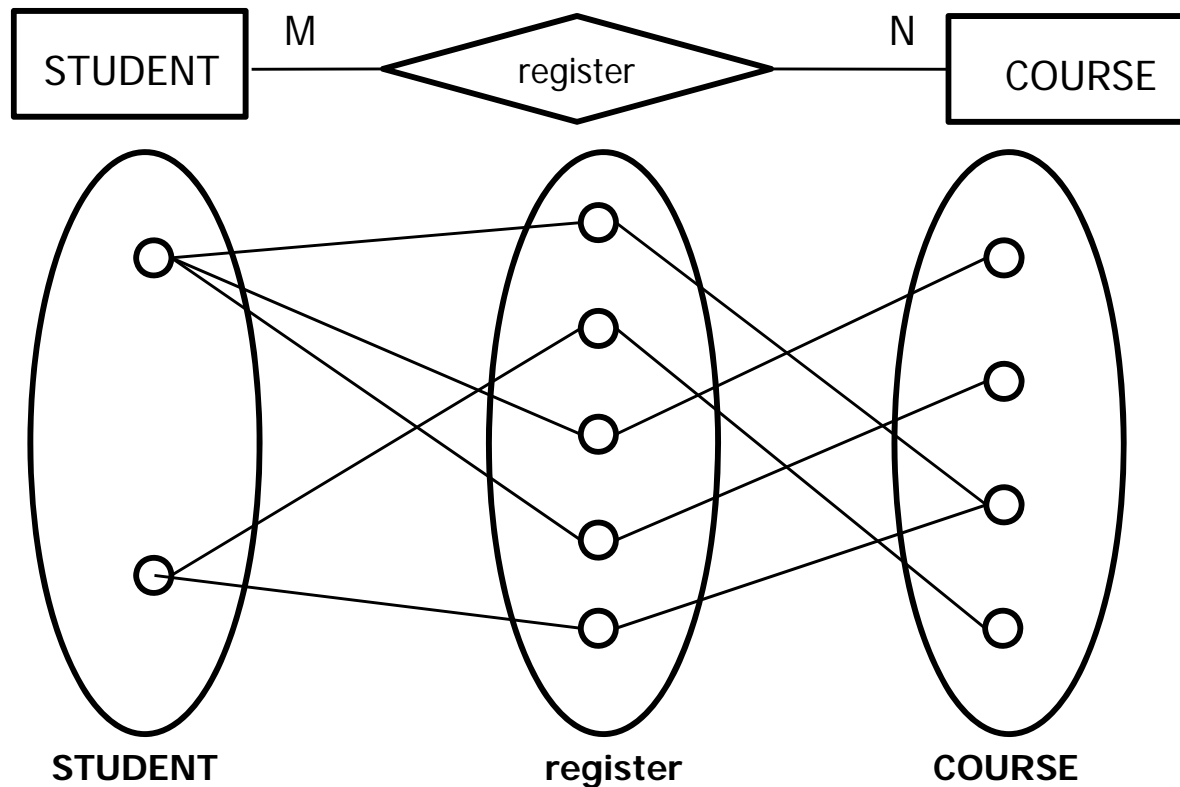
- เอนทิตีหนึ่งตัวในเซตหนึ่งมีความสัมพันธ์กับเซตอื่นมากกว่าหนึ่งตัว



5.3.3 Relationship cardinality

3. Many-to-many relationship M:N

- เอนทิตีแต่ละตัวในเซตแรกสัมพันธ์กับหลายเอนทิตีในเซตที่สอง และ
- เอนทิตีในเซตที่สองแต่ละตัวสัมพันธ์กับเอนทิตีหลายตัวในเซตแรก



5.3.4 Relationship participation

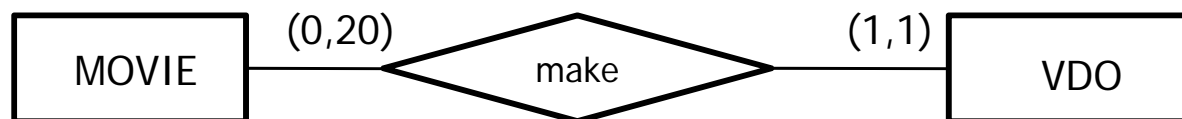
- การมีส่วนร่วมในความสัมพันธ์
บอกลักษณะความสัมพันธ์ที่ละเอียดมากขึ้น ทำให้ทราบจำนวนเอนทิตีต่ำสุด (Minimum cardinality) ที่สัมพันธ์กับอีกเอนทิตีหนึ่ง
- แบ่งเป็น 2 ลักษณะ
 1. ความสัมพันธ์แบบบังคับ (Mandatory relationship)
 - เอนทิตีทุกตัวของเซตต้องมีส่วนร่วมในความสัมพันธ์ของเอนทิตีเซต
 2. ความสัมพันธ์แบบทางเลือก (Optional relationship)
 - เอนทิตีบางตัวไม่มีส่วนร่วมในความสัมพันธ์ของเอนทิตีเซต

5.3.4 Relationship participation

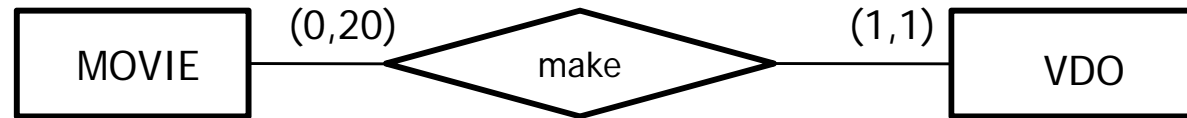
- ตัวอย่าง ร้านให้เช่าวีดีโอเก็บสต็อกของวีดีโอที่ผลิตจากภาพยนตร์ โดยภาพยนตร์หนึ่งเรื่องอาจถูกทำเป็นม้วนวีดีโอเพื่อให้สมาชิกเช่าหลายชุด แต่ภาพยนตร์บางเรื่องที่ไม่ได้รับความนิยมจะไม่ถูกนำไปจัดทำม้วนวีดีโอเลย

ดังนั้นสำหรับภาพยนตร์หนึ่งเรื่อง จำนวนวีดีโอที่ถูกจัดทำจึงมีตั้งแต่ 0 ถึง 20 ม้วนขึ้นอยู่กับความนิยมขณะนั้น

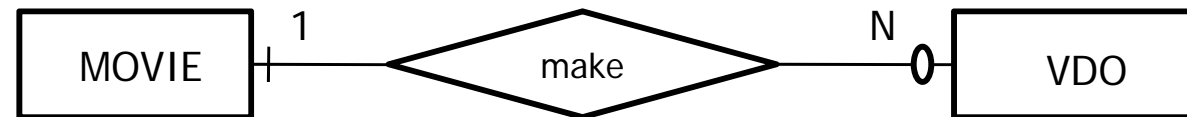
ส่วนวีดีโอ 1 ชุดจะทำจากภาพยนตร์เพียงหนึ่งเรื่องเท่านั้น



5.3.4 Relationship participation

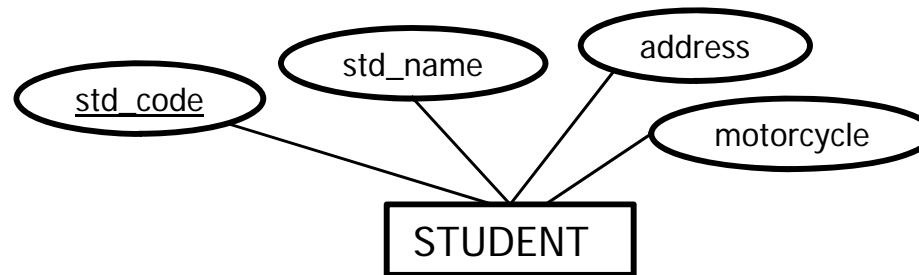


- จะเห็นว่า หนังบางเรื่องไม่มีความสัมพันธ์กับวีดีโอ เรียกว่า เอนทิตี **VDO** มีความสัมพันธ์แบบทางเลือกร่วมกับ **make**
- ขณะเดียวกัน วีดีโอแต่ละม้วนมีความสัมพันธ์กับหนังหนึ่งเรื่อง เรียกว่า เอนทิตี **MOVIE** มีความสัมพันธ์แบบบังคับกับ **make**

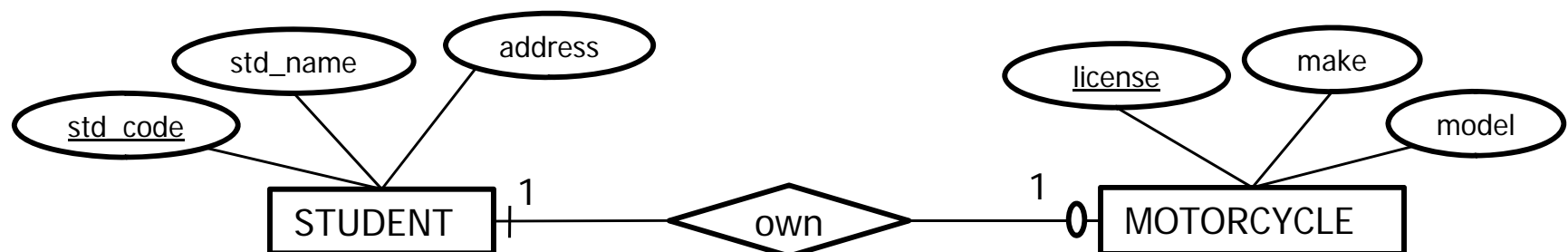


5.3.5 ความสัมพันธ์ที่พัฒนามาจากแอททริบิวต์

- Ex. หน่วยอาคารสถานที่ต้องการเก็บข้อมูลเกี่ยวกับการใช้จักรยานยนต์ของนศ. โดยในฐานข้อมูลจะบันทึกข้อมูลของนศ.ซึ่งประกอบด้วย std_code, std_name, address, motorcycle

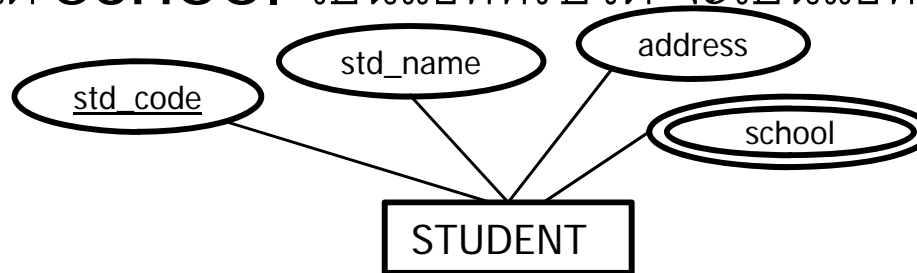


- พบว่า นศ.ส่วนใหญ่ไม่มีรถจักรยานยนต์ ดังนั้นค่าส่วนใหญ่เป็นค่า Null
- นอกจากนี้ยังต้องการเก็บข้อมูล license, make, model
- กรณีเช่นนี้ motorcycle ควรเป็นเอนทิตี มากกว่าแอททริบิวต์

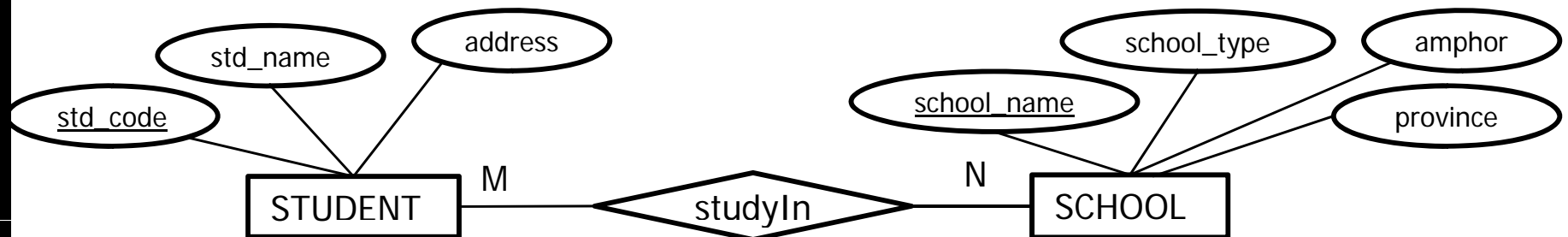


5.3.5 ความสัมพันธ์ที่พัฒนามาจากแอททริบิวต์

- Ex. ต้องการเก็บข้อมูลนักศึกษา โดยเพิ่มเติมเกี่ยวกับการศึกษาระดับมัธยมศึกษา และมีนศ.บางคนที่มีการย้ายร.ระหว่างเรียน
- ถ้ากำหนด **school** เป็นแอททริบิวต์ จะเป็นแอททริบิวต์แบบหลายค่า

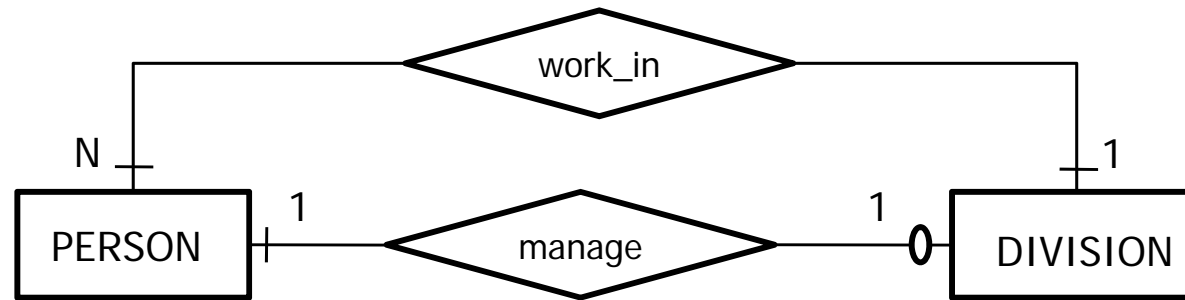


- ข้อมูลร. ประกอบด้วย school_name, school_type, amphor, province จึงแปลงแอททริบิวต์ school เป็นเอนทิตี



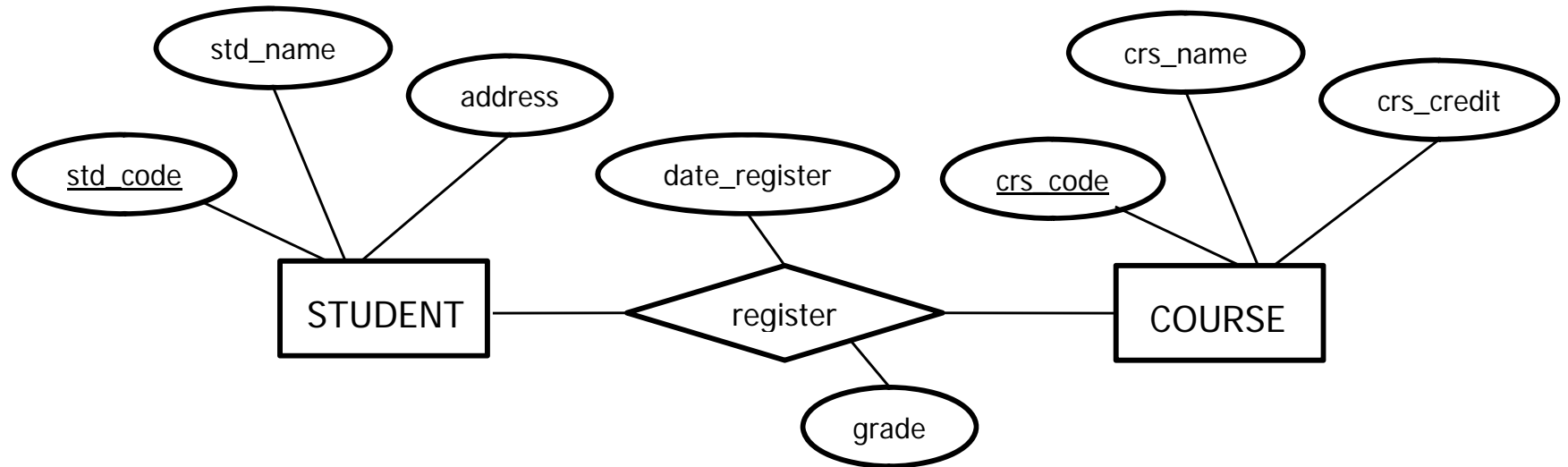
5.3.6 Multiple relationship

- บางครั้งเอนทิตี 2 เอนทิตี อาจจะมีความสัมพันธ์กันมากกว่า 1 แบบ
- พนักงานทำงานในแผนก และพนักงานเป็นผู้จัดการแผนก



5.3.7 การแปลง ERD เป็นตัวแบบฐานข้อมูลเชิงสัมพันธ์

- วิธีการแบ่งจะพิจารณาจากความสัมพันธ์แต่ละรูปแบบ 3 แบบ
- 1. การแปลงความสัมพันธ์แบบ M:N
- สร้างรีเลชันใหม่ที่มีคีย์หลักเป็นคีย์ประกอบจากคีย์หลักของ 2 เอนทิตี
- นำแอททริบิวต์ของความสัมพัทธ์มาบรรจุไว้ด้วย (ถ้ามี)



STUDENT(std_code, std_name, address)

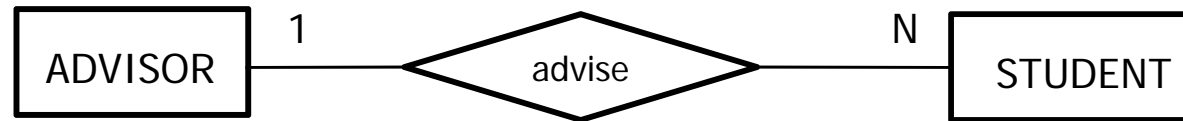
COURSE(crs_code, crs_name, crs_credit)

REGISTRATION(std_code, crs_code, date_register, grade)

5.3.7 การแปลง ERD เป็นตัวแบบฐานข้อมูลเชิงสัมพันธ์

2. การแปลงความสัมพันธ์แบบ 1:N

- ให้นำคีย์หลักของรีเลชันด้าน 1 มาเป็นคีย์นอกของรีเลชันด้าน N



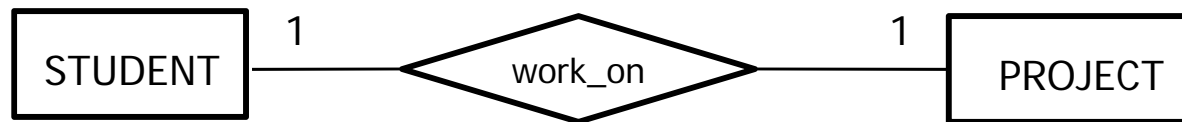
ADVISOR(adv_code, adv_name, office_phone)

STUDENT(std_code, std_name, address, adv_code)

5.3.7 การแปลง ERD เป็นตัวแบบฐานข้อมูลเชิงสัมพันธ์

3. การแปลงความสัมพันธ์แบบ 1:1

- แปลงเอนทิตีแต่ละตัวเป็นรีเลชัน
- นำคีย์หลักของรีเลชันด้านหนึ่งไปเป็นคีย์นอกของอีกรีเลชันหนึ่ง โดยเลือกทำเพียงด้านเดียวเท่านั้น



STUDENT(std_code, std_name, address)

PROJECT(proj_no, proj_title, std_code)

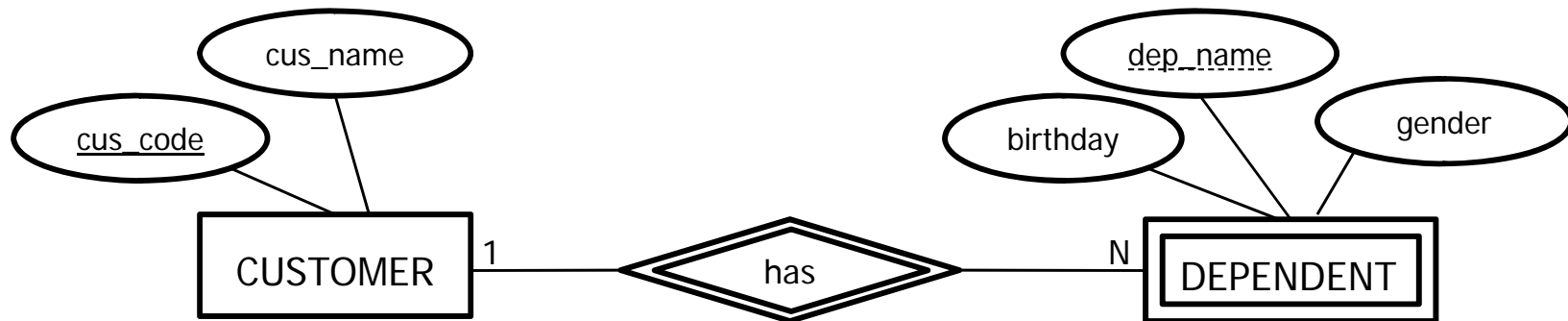
5.4 Weak entity type

- เอนทิตีเซตแบ่งได้เป็น 2 กลุ่ม
 1. เอนทิตีที่สามารถเกิดขึ้นโดยอิสระ และมีตัวชี้เฉพาะเพื่อบ่งชี้เอนทิตี
 - เช่น STUDENT มี std_code เป็นตัวชี้เฉพาะ
 - Strong entity หรือ Independent entity
 2. เอนทิตีที่ไม่สามารถเกิดขึ้นได้ด้วยตนเอง ต้องพึ่งพาเอนทิตีอื่น
 - มักจะไม่มีตัวชี้เฉพาะของตนเอง
 - เอนทิตีแบบอ่อน / Weak entity / Dependent entity

5.4.1 Weak entity type

- Ex. บริษัทประกันชีวิตต้องการสร้างฐานข้อมูลลูกค้า ซึ่งต้องการทำประกันชีวิตเพื่อคุ้มครองบุตร โดยแต่ละกรมธรรม์ต้องระบุผู้ที่รับการคุ้มครอง ถ้าลูกค้ายกเลิกการทำประกัน กรมธรรม์ก็จะสิ้นสุด บริษัทจะลบข้อมูลทุกอย่างที่เกี่ยวข้องกับลูกค้า
- จะเห็นว่าข้อมูลบุตร (Dependent) จะไม่สามารถเกิดขึ้นได้ หากไม่มีข้อมูลลูกค้า (Customer)
- เรียก CUSTOMER ว่า เอนทิตีแม่ (Parent, Owner, Identifying entity) ของเอนทิตีเซต DEPENDENT
- เรียก DEPENDENT ว่า เอนทิตีลูก (Child entity)

5.4.1 Weak entity type



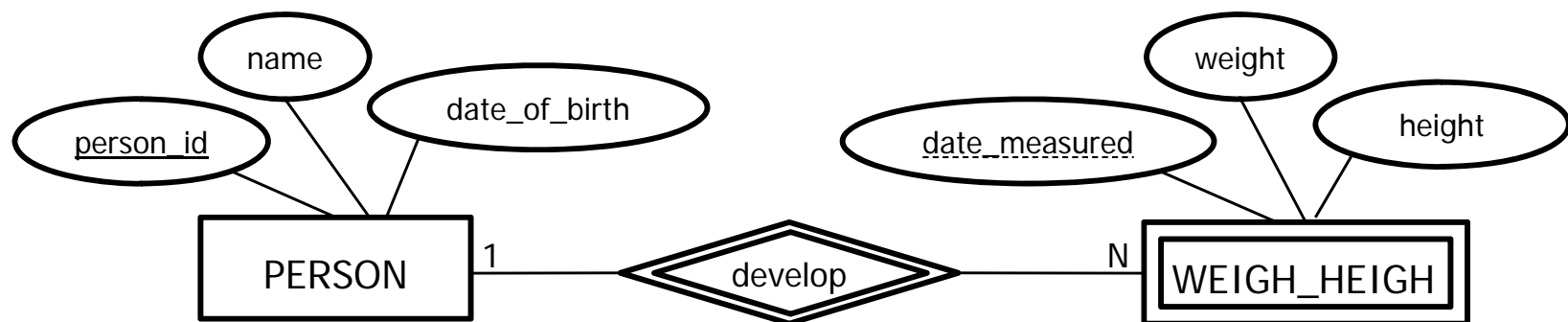
- เอนทิตีแบบอ่อน มีแอททริบิวต์เหมือนเอนทิตีปกติ
- ยกเว้นไม่มีแอททริบิวต์ชี้เฉพาะ
- เอนทิตีแบบอ่อนจะมีเอนทิตีแม่หนึ่งตัวเสมอ
- ความสัมพันธ์มักจะเป็นแบบ 1:N
- dep_name มีสิทธิ์ซ้ำกันได้ จึงไม่ใช่ตัวชี้เฉพาะ
- dep_name ในกลุ่มที่มีแอททริบิวต์แม่เดียวกัน จะไม่ซ้ำกัน ลักษณะนี้เรียกว่า Partial identifier ใน ERD แสดงด้วย เส้นประ

5.4.2 การแปลงเอนทิตีแบบอ่อนเป็นรีเลชัน

- ให้แปลงเอนทิตีแบบอ่อนเป็น 1 รีเลชัน
- นำคีย์หลักของเอนทิตีแม่มาผสมกับ Partial identifier ของเอนทิตีอ่อน
CUSTOMER(cus_code, cus_name)
DEPENDENT(dept_name, cus_code, birthday, gender)
- การพิจารณาว่า เอนทิตีเป็นเอนทิตีแบบอ่อน
 - ในเอนทิตีมีตัวชี้เฉพาะเป็นเอกลักษณ์ของตัวเองหรือไม่
 - ถ้าตัดเอนทิตีที่สัมพันธ์กันออกจากระบบแล้ว เอนทิตีนั้นหายไปด้วย

5.4.3 การสร้างตัวแบบสำหรับข้อมูลที่แปรตามกาลเวลา (Attribute variations)

- เอนทิตีแบบอ่อนสามารถนำมาประยุกต์ใช้ เมื่อแอททริบิวต์บางตัวมีค่าแปรเปลี่ยนตามเวลา (Time-dependent attribute)
- อาจเรียกว่า Historical attribute
- Ex. สถานเลี้ยงเด็กกำพร้าต้องการติดตามการพัฒนาสุขภาพของเด็ก โดยสร้างฐานข้อมูลเกี่ยวกับความสูง และน้ำหนักของเด็กที่เปลี่ยนแปลงแต่ละเดือน

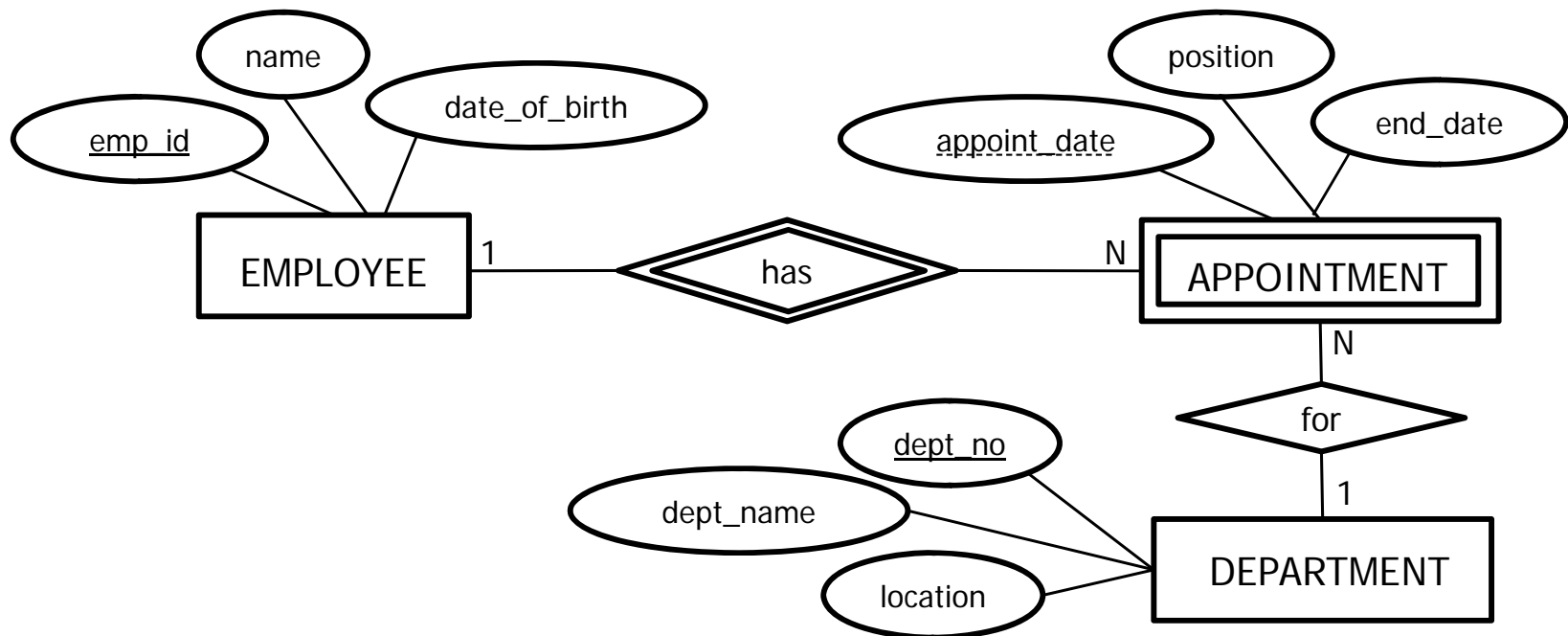


PERSON(person_id, name, date_of_birth)

WEIGH_HEIGHT(person_id, date_measured, weight, height)

5.4.3 การสร้างตัวแบบสำหรับข้อมูลที่แปรตามกาลเวลา (Attribute variations)

- Ex.ฐานข้อมูลเพื่อติดตามข้อมูลการแต่งตั้งพนักงานให้มีตำแหน่งสูงขึ้น โดยพนักงานที่ปรับเปลี่ยนตำแหน่งอาจทำงานที่แผนกเดิมหรือย้ายแผนก



DEPARTMENT(dept_no,dept_name,location)

EMPLOYEE(emp_id,name,date_of_birth)

APPOINTMENT(emp_id,appoint_date,position,end_date,dept_no)



5.5 Degree of relationship

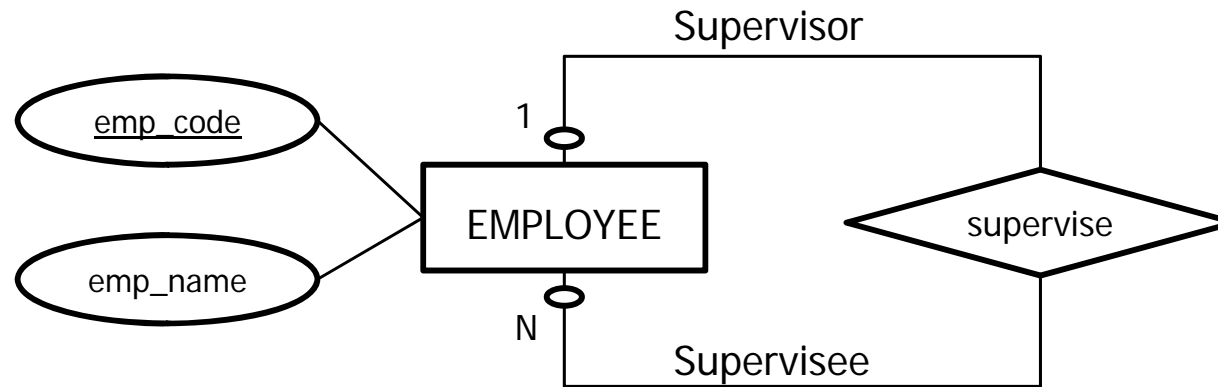
- ดีกรีของความสัมพันธ์
จำนวนเอนทิตีเซตที่มีส่วนร่วมในเซตของความสัมพันธ์หนึ่ง
- มี 3 ประเภท ตามจำนวนดีกรี
 1. ความสัมพันธ์ดีกรี 1 หรือ **Unary relationship**
ความสัมพันธ์ที่เกิดขึ้นในเอนทิตีเซตเดียวกัน
 2. ความสัมพันธ์ดีกรี 2 หรือ **Binary relationship**
ความสัมพันธ์ที่เกิดขึ้นระหว่าง 2 เอนทิตีเซต
 3. ความสัมพันธ์ดีกรี 3 หรือ **Ternary relationship**
ความสัมพันธ์ที่เกิดขึ้นระหว่าง 3 เอนทิตีเซต



5.5.1 Unary relationship

- อาจเรียกว่า ความสัมพันธ์กับตัวเอง (Recursive relationship)
- มีเอนทิตีเซตเดียวที่มีส่วนร่วมในความสัมพันธ์
- เกิดจากให้เอนทิตีเซตเดียวมีบทบาทในความสัมพันธ์ที่ต่างกัน เช่น
- ต้องการติดตามข้อมูลพนักงานในบริษัท ในด้านการทำงานเป็นที่ปรึกษาให้กับพนักงานใหม่ที่อยู่ภายในแผนกเดียวกัน โดยมีนโยบายว่า
- พนักงานทุกคนที่มีอายุงานน้อยกว่า 2 ปี ต้องอยู่ภายใต้การดูแลของที่ปรึกษาหนึ่งคน
- จะเห็นว่าทั้งพนักงานใหม่และที่ปรึกษาต่างเป็นพนักงานทั้งคู่ ซึ่งเก็บข้อมูลอยู่ในเอนทิตีเซตเดียวกัน คือ EMPLOYEE

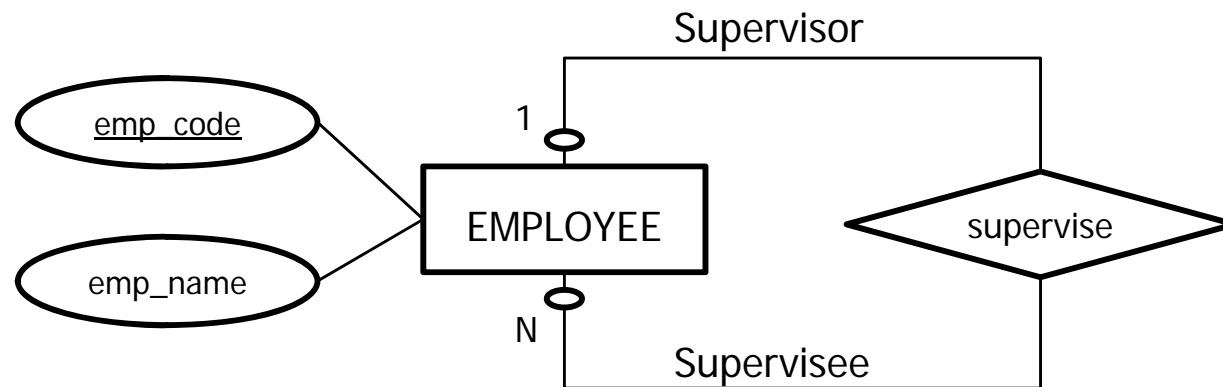
5.5.1 Unary relationship



- เพื่อเน้นถึงบทบาทที่แตกต่างกัน อาจกำหนด **Role name** ได้
- การแปลงความสัมพันธ์แบบยูนารีเป็นรีเลชัน มี 2 แบบ
 - หากเป็นความสัมพันธ์แบบหนึ่งต่อหนึ่ง หรือหนึ่งต่อหลาย
 - หากเป็นความสัมพันธ์แบบหลายต่อหลาย

5.5.1 Unary relationship

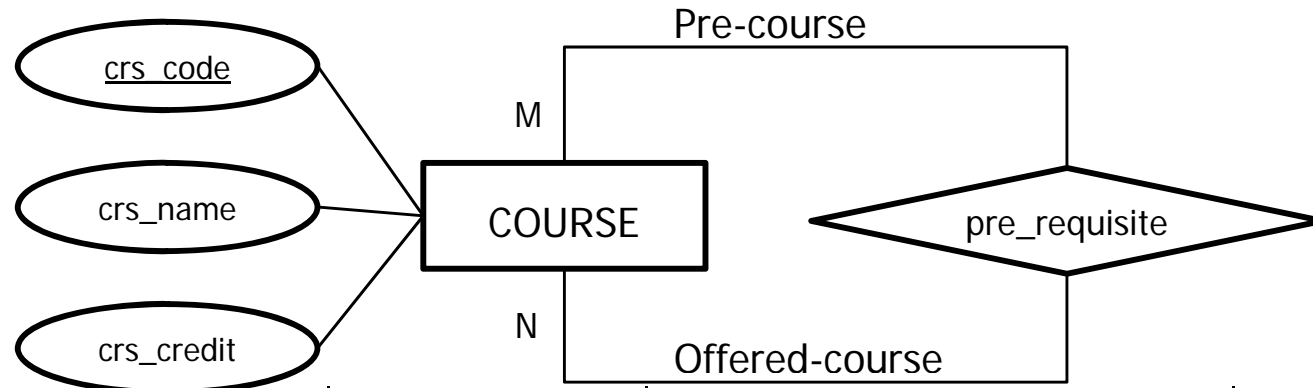
- กรณีความสัมพันธ์แบบหนึ่งต่อหนึ่ง หรือหนึ่งต่อหลาย



- แปลงเอนทิตีเซตเป็นหนึ่งรีเลชัน
- เพิ่มคีย์นอก ซึ่งอ้างอิงคีย์หลักเข้าไปในรีเลชันเดียวกันนี้ เช่น
EMPLOYEE(emp_code, emp_name, supervisor_code)
- supervisor_code เป็น Recursive FK ที่มีโดเมนเดียวกันคีย์หลัก

5.5.1 Unary relationship

- กรณีความสัมพันธ์แบบหลายต่อหลาย



- กระบวนวิชาหนึ่งอาจต้องมีวิชาที่ผ่านก่อนหลายวิชา และวิชาหนึ่งอาจเป็นวิชาที่ต้องเรียนก่อนหลายวิชา
- บทบาทของวิชา มี 2 ลักษณะ คือ วิชาที่เปิดสอน กับ วิชาที่ต้องเรียนก่อน
- แปลง COURSE เป็นรีเลชัน
- สร้างรีเลชันใหม่สำหรับความสัมพันธ์ pre_requisite และบรรจุคีย์หลักของเอนทิตี COURSE ที่เกี่ยวข้องกับทั้งสองบทบาท โดยตั้งชื่อต่างกัน

COURSE(crs_code, crs_name, crs_credit)

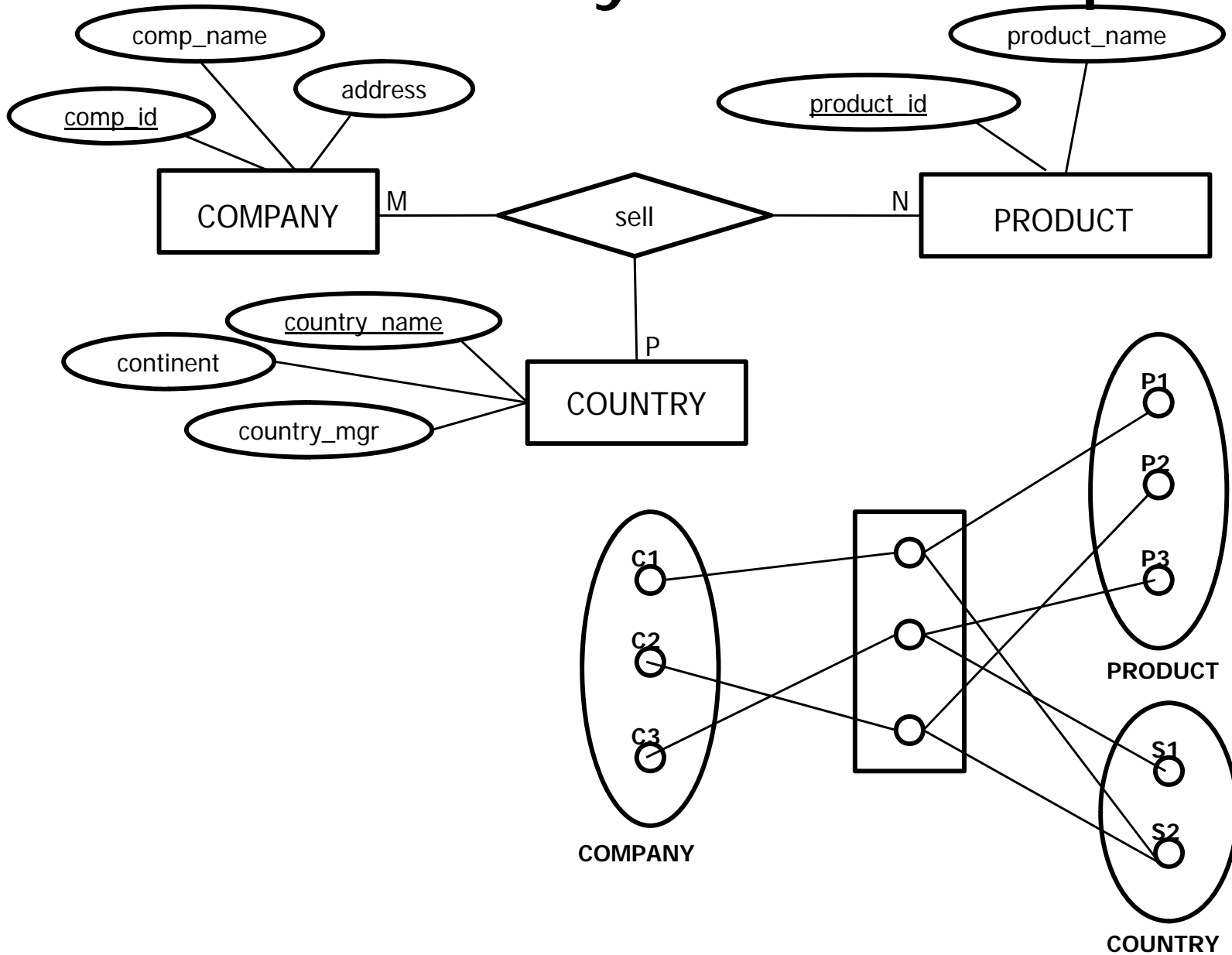
PREREQUISITE(crs_code, pre_crsCode)



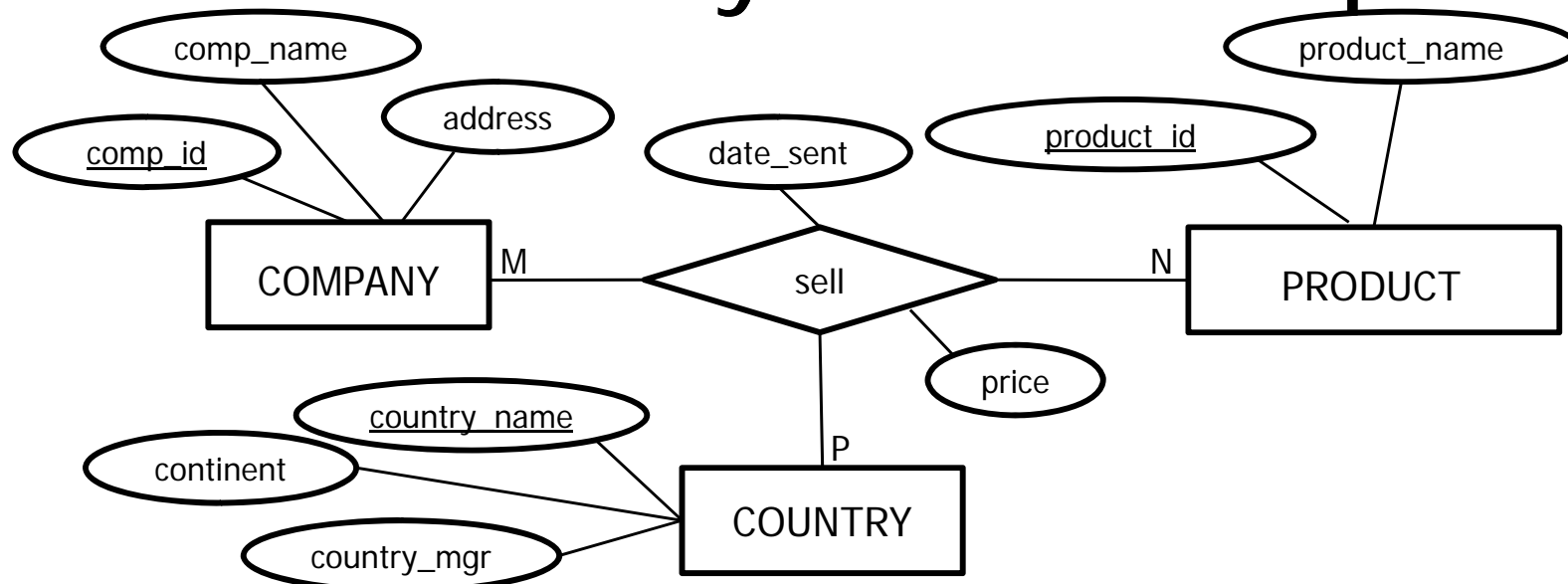
5.5.3 Ternary relationship

- ความสัมพันธ์ที่เกิดจากเอนทิตีมากกว่า 2 ตัว เรียกว่า n-ary relationship
- เมื่อ n คือ จำนวนเอนทิตีในความสัมพันธ์
- กรณี Ternary คือ 3-ary มี 3 เอนทิตีในความสัมพันธ์ เช่น
- ความสัมพันธ์ของการที่บริษัทส่งสินค้า ไปจำหน่ายยังประเทศต่าง ๆ ทั่วโลก แต่ครั้งของการส่งสินค้าจะบันทึกวันที่ส่งสินค้า และราคาขาย โดยที่สินค้าชนิดเดียวกันเมื่อส่งไปขายยังประเทศต่างกันอาจมีราคาต่างกัน แต่ละบริษัทส่งสินค้าไปบางประเทศ ขึ้นอยู่กับประเภทสินค้า

5.5.3 Ternary relationship



5.5.3 Ternary relationship



- กฎการแปลงเป็นรีเลชัน

1. แปลงแต่ละเอนทิตีเป็น 1 รีเลชัน

COMPANY(comp_id,comp_name,address)

PRODUCT(product_id,product_name)

COUNTRY(country_name,continent,country_mgr)

2. แปลงความสัมพันธ์เป็นรีเลชันใหม่ มีคีย์หลักจากคีย์หลักของรีเลชันในข้อ 1

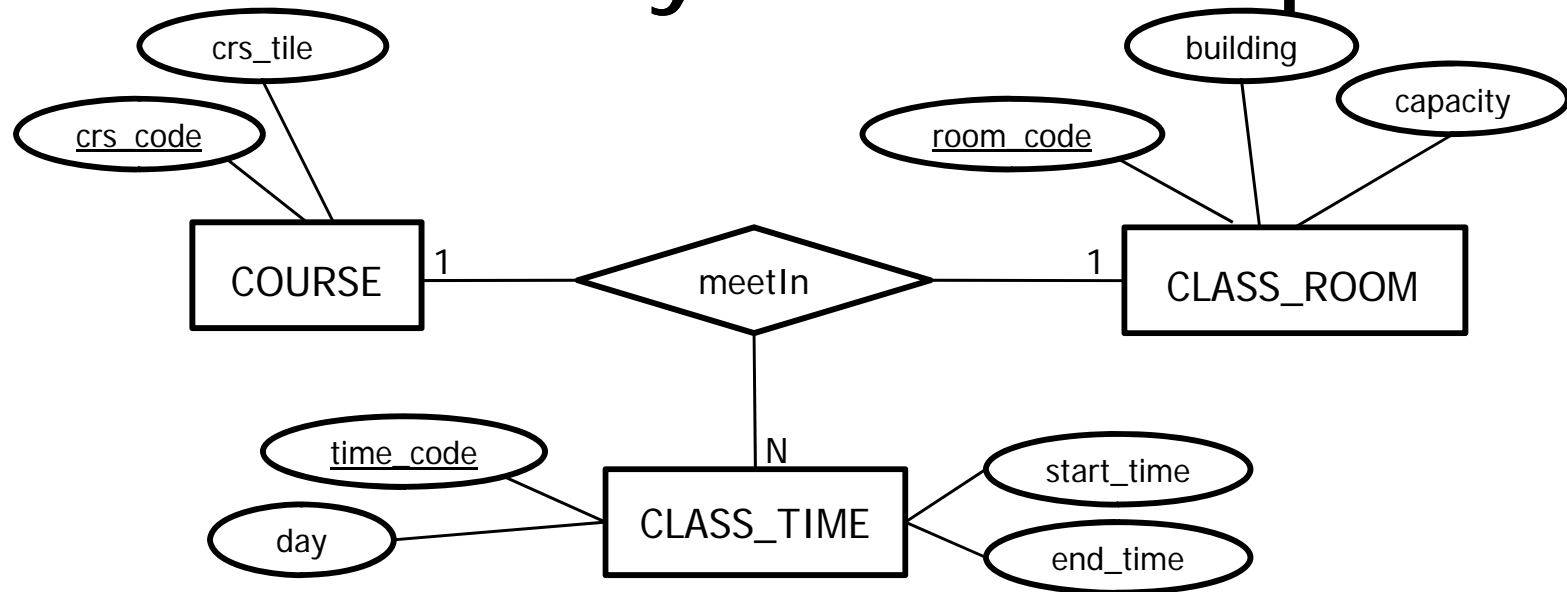
SELL(comp_id,product_id,country_name,date_sent,price)



5.5.3 Ternary relationship

- Ex. ความสัมพันธ์ระหว่างวิชา ห้องเรียน และเวลาเรียน
- วิชาหนึ่งแยกเป็นหลายตอนมีเวลาสอนที่ต่างกัน แต่ ณ เวลาหนึ่งจะใช้ห้องเรียนเพียงห้องเดียวเท่านั้น
- วิชาเดียวกันอาจใช้ห้องเรียนเดียวกันหรือไม่ก็ได้ถ้าเรียนเวลาต่างกัน
- ถ้าระบุเวลาเรียนและห้องเรียน สามารถบอกได้ว่าวิชาที่สอนเป็นวิชาอะไร
- ถ้าระบุเวลาเรียนและชื่อวิชาก็สามารถบอกห้องเรียนได้
- แต่ถ้าระบุเวลาเรียน จะบอกไม่ได้ว่าวิชาไหน
- ถ้าระบุห้องเรียนและวิชา ก็บอกเวลาเรียนไม่ได้

5.5.3 Ternary relationship



- แปลงเป็นรีเลชัน

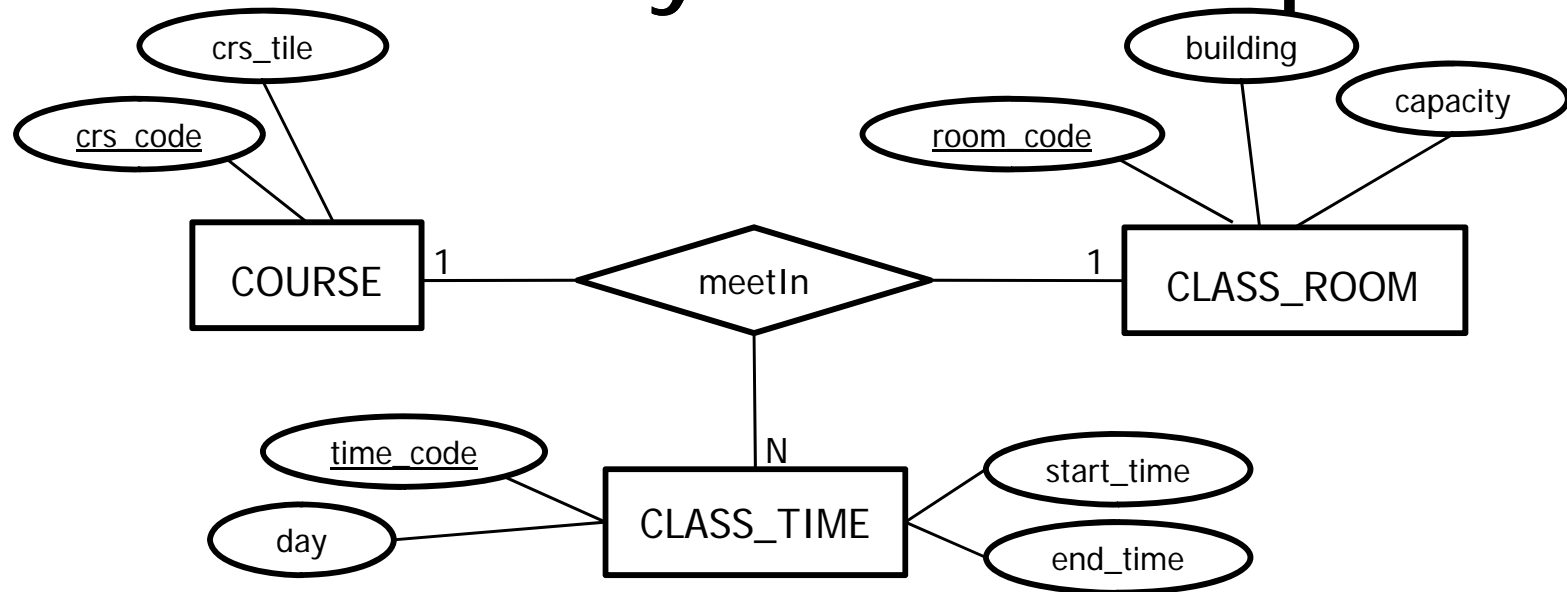
COURSE(crs_code, crs_title)

CLASS_ROOM(room_code, building, capacity)

CLASS_TIME(time_code, day, start_time, end_time)

CLASS_MEETING(crs_code, time_code, room_code)

5.5.3 Ternary relationship



- การขึ้นต่อกันแบบฟังก์ชันของ CLASS_MEETING

$FD_1 : crs_code, time_code \rightarrow room_code$

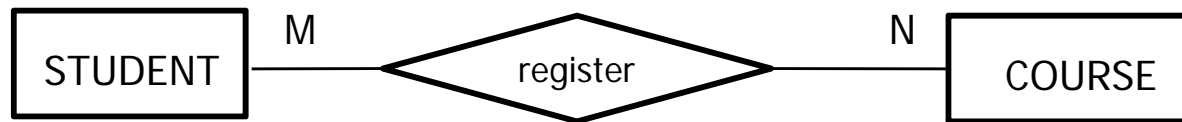
$FD_2 : time_code, room_code \rightarrow crs_code$

- ดังนั้น Candidate key คือ $\{crs_code, time_code\}$ และ $\{time_code, room_code\}$

- ข้อสังเกตคือ คาร์ดินัลลิตีของ n-ary relationship อาจเป็น 1:1:1, 1:1:M, 1:M:N, M:N:P

5.6 Composite or Associative entity

- เอนทิตีประกอบ
เป็นเอนทิตีที่นำตัวชี้เฉพาะของเอนทิตีอื่น 2 ตัวขึ้นไปมาประกอบกันเป็นตัวชี้เฉพาะของตัวเอง มักจะมาจากความสัมพันธ์แบบ M:N
 - แปลงเป็นความสัมพันธ์ M:N เป็น 1:N และ N:1 เช่น



- นศ. 1 คน ลงทะเบียนได้หลายครั้ง แต่ครั้งของการลงทะเบียนเป็นของนศ. 1 คน
- การลงทะเบียน 1 ครั้งสำหรับ 1 วิชา วิชาหนึ่งถูกลงทะเบียนได้หลายครั้ง

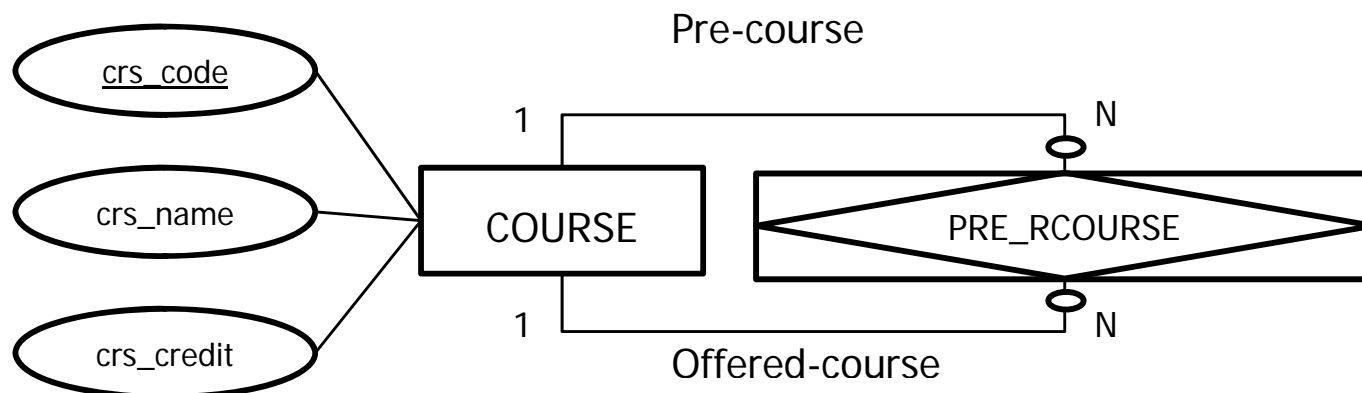
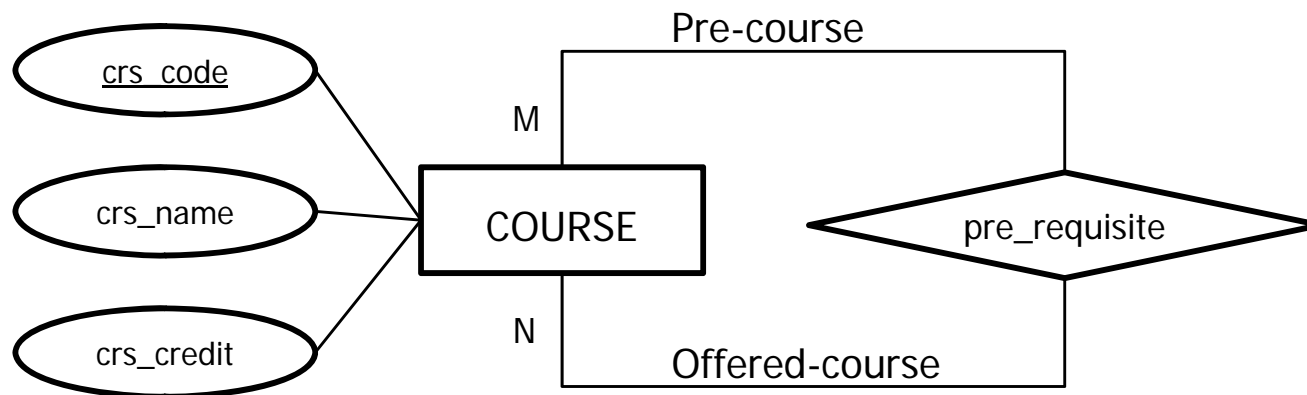


5.6 Composite or Associative entity

- การกำหนดตัวชี้เฉพาะของเอนทิตีประกอบ ทำได้ 2 วิธี
 1. นำตัวชี้เฉพาะของเอนทิตีเดิมมารวมกัน
 - จากตัวอย่าง REGISTRY จะได้ {std_code, crs_code}
 - เรียกว่า Default identifier
 2. สร้างตัวชี้เฉพาะขึ้นมาใหม่ เรียกว่า Surrogate identifier
 - ผู้ออกแบบสามารถกำหนดตัวชี้เฉพาะขึ้นมาใหม่ในกรณีพบว่า Default identifier มีโอกาสซ้ำกัน หรือมีตัวชี้เฉพาะที่ผู้ใช้คุ้นเคยอยู่แล้ว เช่น
 - กำหนด reg_no เพื่อแทนลำดับการลงทะเบียนของนักศึกษา เป็นตัวชี้เฉพาะ

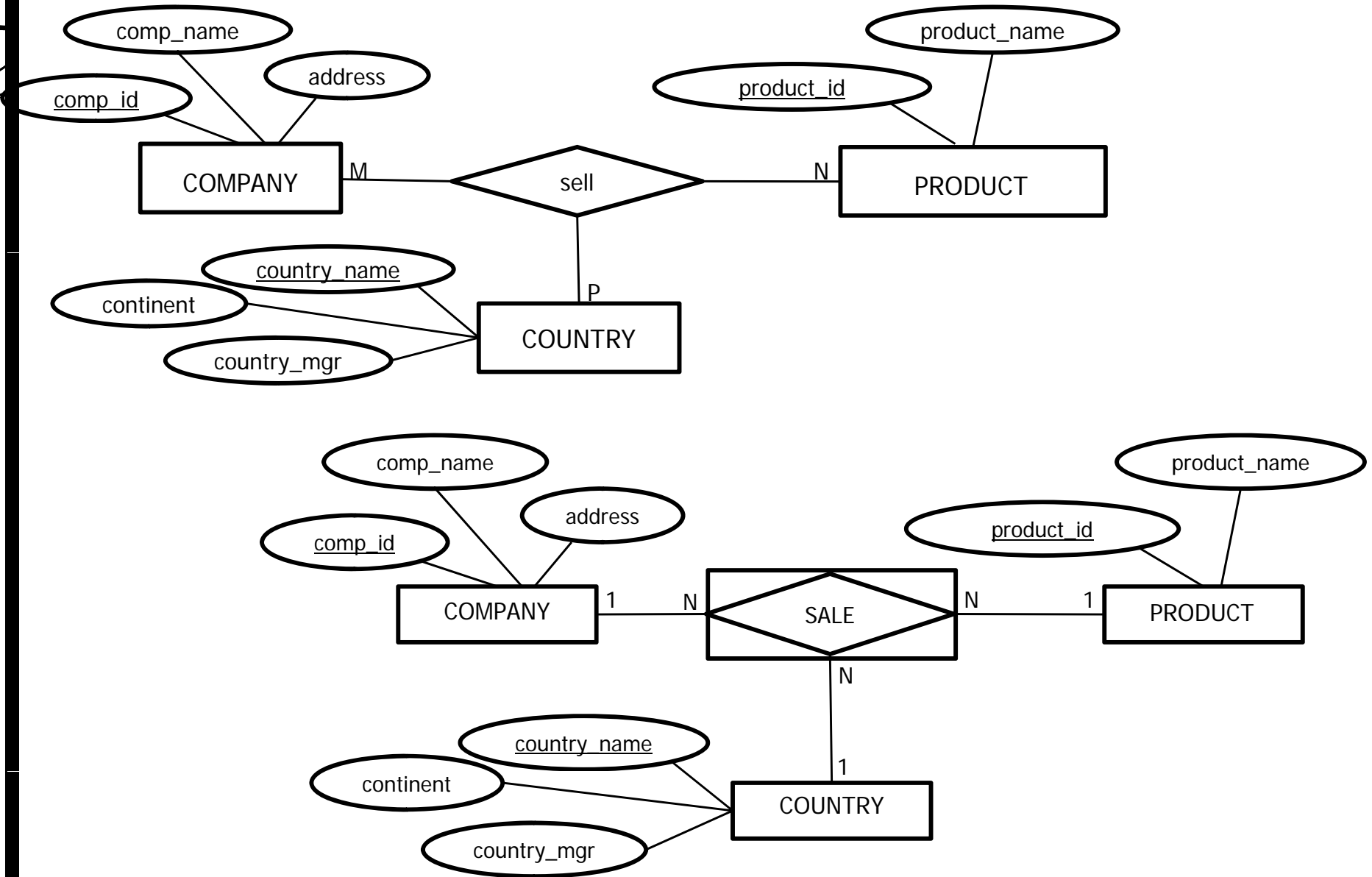
5.6 Composite or Associative entity

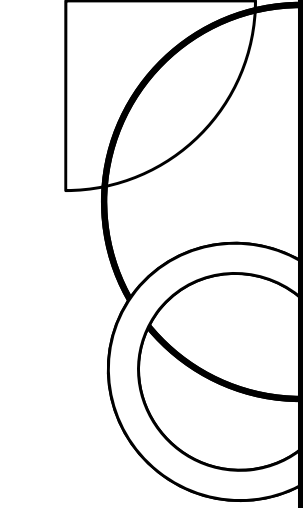
- เอนทิตีประกอบจะเกิดกับความสัมพันธ์ที่คาร์ดินัลลิตีเป็นแบบ Many เช่น
- ปัญหาวิชาที่ต้องลงทะเบียนก่อน เป็น Unary relationship



5.6 Composite or Associative entity

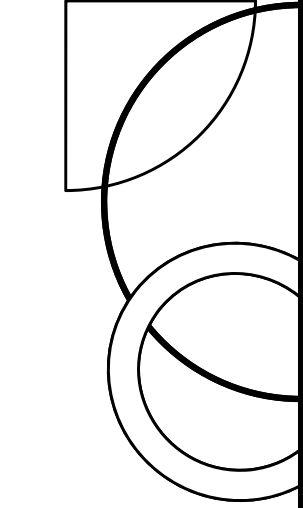
- ปัญหาขายสินค้า เป็น Ternary-relationship





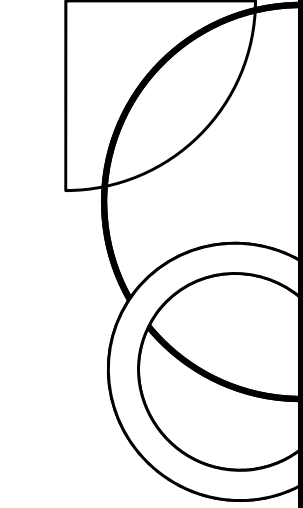
5.7 กฎธุรกิจ

- การพิจารณาข้อมูลอย่างเดียว ไม่สามารถทำให้เข้าใจธุรกิจขององค์กรได้
- การพัฒนาตัวแบบข้อมูลให้ถูกต้อง จะต้องทำความเข้าใจกฎเกณฑ์การทำงานหรือระเบียบปฏิบัติที่กำหนดขึ้นในองค์กรนั้น ๆ
- ซึ่งแตกต่างกันไปตามองค์กร เรียกว่า กฎธุรกิจ
- กฎธุรกิจ จะให้ข้อมูลเกี่ยวกับการดำเนินงานของธุรกิจ (Business metadata) ทำให้เกิดการปฏิบัติหรือดำเนินงานตามแนวทางที่กำหนด เช่น
- นักศึกษาทุกคนในมหาวิทยาลัยต้องมีอาจารย์ที่ปรึกษาหนึ่งคน ทำให้ข้อมูลนักศึกษาสัมพันธ์กับข้อมูลอาจารย์



5.7 กฎธุรกิจ

- กฎธุรกิจบางกฎไม่สามารถแสดงด้วยตัวแบบข้อมูล ตัวอย่าง ให้หาว่ากฎข้อใดไม่เกี่ยวข้องกับตัวแบบข้อมูล
- นักศึกษาจะลงทะเบียนวิชาหนึ่งได้ก็ต่อเมื่อได้สอบผ่านวิชาที่ต้องผ่านก่อนของวิชานั้นแล้ว
- ลูกค้าของสหกรณ์จะเป็นเจ้าของบัญชีเงินฝากได้เพียงบัญชีเดียว และแต่ละบัญชีต้องมีเจ้าของเพียงคนเดียว
- ให้พนักงานทุกคนแต่งกายในชุดประจำชาติทุกวันศุกร์
- ในแต่ละเทอมอาจารย์หนึ่งคนต้องสอนบรรยายอย่างน้อย 1 วิชา แต่ไม่เกิน 3 วิชา และในแต่ละวิชาต้องมีผู้สอนไม่เกิน 4 คน



5.7 กฎธุรกิจ

- แหล่งที่มาของกฎธุรกิจ คือ
 - ผู้จัดการบริษัท ผู้กำหนดนโยบายของบริษัท
 - ผู้จัดการแผนกหรือผู้บริหารระดับต่าง ๆ
 - เอกสารที่จัดทำขึ้นโดยบริษัท
 - ผู้ใช้ปลายทางขององค์กรนั้น
- มุมมองของผู้ใช้ต่อกฎธุรกิจ บางครั้งอาจจะเป็นเพียงส่วนเดียวของกฎธุรกิจเท่านั้น หากเป็นไปได้ควรมีการตรวจสอบจากหลาย ๆ แหล่งหรือจากเอกสารที่เป็นลายลักษณ์อักษร

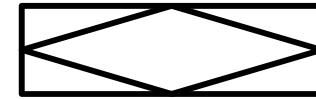
5.8 E-R model notation



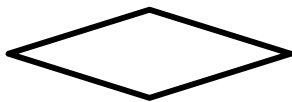
Regular or Strong entity



Weak entity



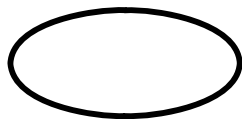
Composite entity



Relationship



Identifying relationship



Attribute



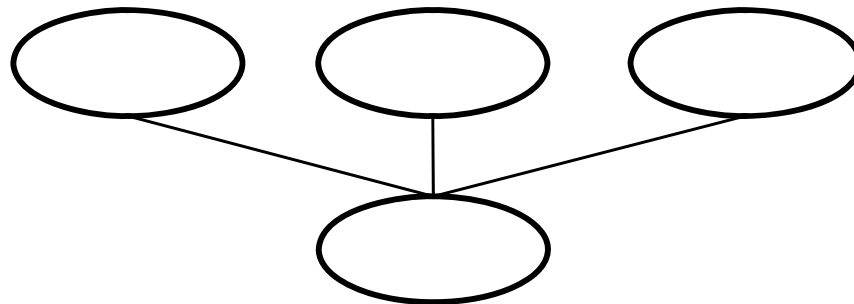
Multi-valued attribute



Identifier or Key attribute



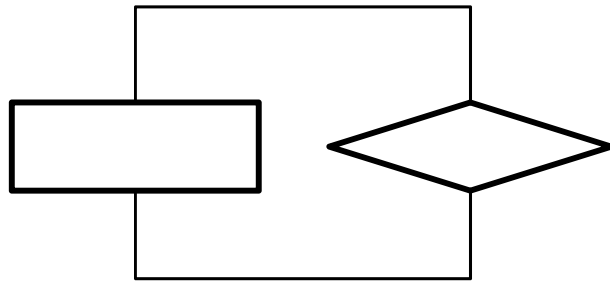
Derived attribute



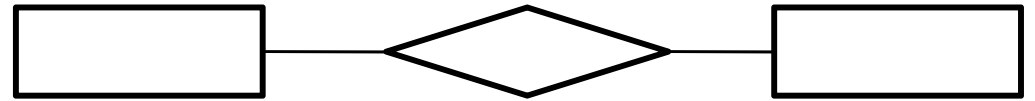
Composite attribute

สัญลักษณ์พื้นฐานสำหรับ ERM

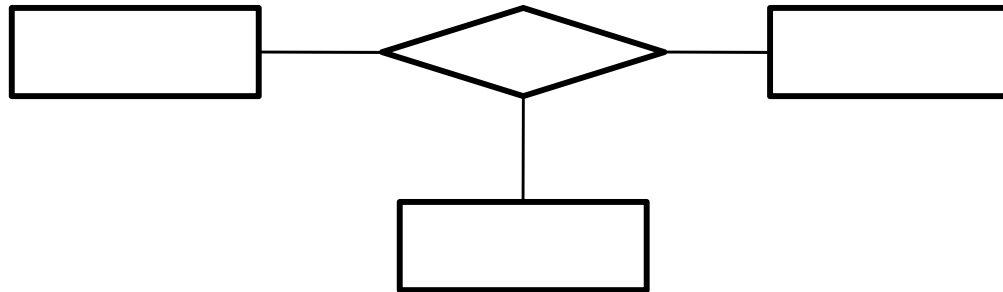
5.8 E-R model notation



Unary relationship



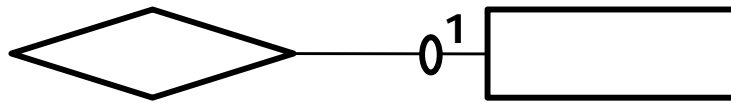
Binary relationship



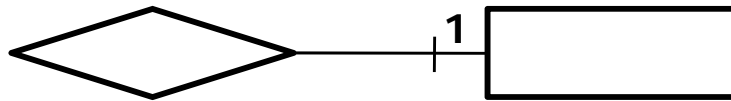
Ternary relationship

สัญลักษณ์แสดงตึกิริของความสัมพันธ์

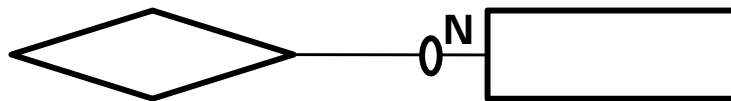
5.8 E-R model notation



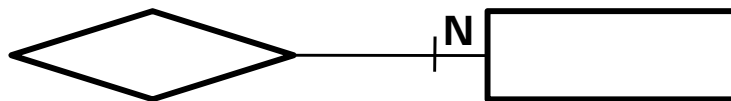
หนึ่งแบบทางเลือก



หนึ่งแบบบังคับ



หลายแบบทางเลือก



หลายแบบบังคับ

สัญลักษณ์แสดงการมีส่วนร่วมของความสัมพันธ์



Exercise

งานเดี่ยว 4, 5, 6

งานเดี่ยว เลือกทำ 1 ข้อ 7, 8, 9

งานกลุ่ม 4 กลุ่ม 10, 11, 12, 13

- ให้เขียนกฎธุรกิจแต่ละข้อพร้อม ERD ส่วนนั้น
- ให้เขียน ERD รวมของระบบ
- ให้เพิ่มกฎธุรกิจได้เท่าที่จำเป็นและไม่ขัดแย้งกับโจทย์
- ให้เขียนโดยไม่ต้องระบุ composite entity
- หากโจทย์ไม่กำหนด cardinality ให้ระบุ 1-1, 1-N, M-N



Exercise

4. จงสร้างผังอีอาร์จากข้อกำหนดเบื้องต้นต่อไปนี้

"ต้องการสร้างฐานข้อมูลเพื่อบันทึกข้อมูลของศูนย์การค้าในประเทศไทย โดยสามารถค้นข้อมูลของศูนย์การค้า ร้านค้า และเจ้าของร้านค้าแต่ละแห่ง ภายในศูนย์การค้าตลอดจนพนักงานของร้านค้าและประเภทของร้านค้าได้"

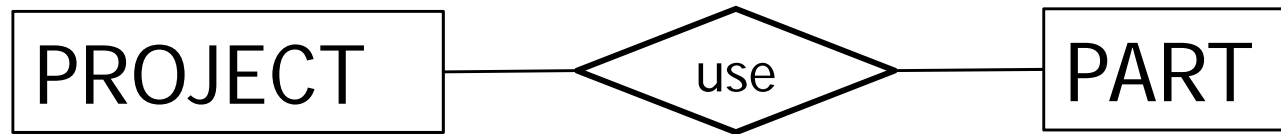


Exercise

- กำหนดกฎธุรกิจต่อไปนี้ จงสร้างผังอีอาร์สำหรับความสัมพันธ์ในแต่ละข้อ
 - ทีมฟุตบอลหนึ่งทีมต้องมีผู้เล่นอย่างน้อย 20 คน
 - แต่ละคณะในมหาวิทยาลัยประกอบด้วยหลายภาควิชา แต่ภาควิชาหนึ่งต้องอยู่ภายใต้คณะเดียว
 - แต่ละภาควิชาต้องมีหัวหน้าภาควิชาบริหารงานหนึ่งคนเสมอ
 - ลูกค้าอาจสั่งซื้อสินค้าจากบริษัทหลายครั้ง แต่การสั่งซื้อของแต่ละครั้งต้องมาจากลูกค้าเพียงรายเดียว
 - ภาควิชาที่เปิดให้นักศึกษามาลงทะเบียนเรียนแบ่งเป็นหลายตอน โดยแต่ละตอนจะกำหนดวันและเวลาสอนแตกต่างกัน

Exercise

6. จงเขียน Occurrence diagram



แต่ละ **project** สามารถใช้ **part** ได้หลายชนิด และ
part แต่ละชนิดถูกใช้ในหลาย **project**



Exercise

7. หอพักของมหาวิทยาลัยต้องการสร้างฐานข้อมูล เพื่อบันทึกข้อมูลนักศึกษาที่พักรักษาในหอพัก โดยติดตามได้ว่านักศึกษาพักอยู่ในห้องใด และติดตามกิจกรรมต่าง ๆ ของหอพักที่จัดทำในแต่ละปีการศึกษา โดยมีรายละเอียดดังนี้
- แต่ละปีนักศึกษา 1 คน สามารถเข้าพักรักษาได้เพียง 1 แห่งเท่านั้น
 - หอพักแต่ละแห่งมีนักศึกษาที่พักรักษาในหอพักนั้น ๆ เป็นคณะกรรมการบริหารหอพักในแต่ละปี
 - แต่ละปีมหาวิทยาลัยแต่งตั้งให้เจ้าหน้าที่หนึ่งคนเป็นแม่บ้านดูแลแต่ละหอพัก และเจ้าหน้าที่ 1 คน เป็นแม่บ้านของหอพักได้แห่งเดียวเท่านั้น
 - การทำกิจกรรมของหอพักต้องมีการวางแผนล่วงหน้าในแต่ละปี ข้อมูลกิจกรรมประกอบด้วย รหัสกิจกรรม ชื่อกิจกรรม เดือนที่จะจัดกิจกรรมงบประมาณค่าใช้จ่ายที่ต้องการ และผู้รับผิดชอบกิจกรรมซึ่งเป็นกลุ่มของนักศึกษาที่พักรักษาในหอพักนั้น

Exercise

8. คณะวิทยาศาสตร์และเทคโนโลยีประกอบด้วยภาควิชา 10 ภาควิชา แต่ละภาควิชามีห้องปฏิบัติการคอมพิวเตอร์ที่มีเครื่องคอมพิวเตอร์อย่างน้อย 20 เครื่องขึ้นไปสำหรับให้บริการนักศึกษา โดยแต่ละห้องปฏิบัติการจะเปิดให้บริการนักศึกษาในช่วงเวลาต่างกัน เช่น

ห้อง PB333 เปิดระหว่างเวลา 9.30-12.00 น.

ห้อง MB661 เปิดระหว่างเวลา 10.00-18.00 น.

คณะวิทยาศาสตร์ต้องการสร้างระบบฐานข้อมูลเพื่อบันทึกข้อมูลเกี่ยวกับ

- ภาควิชา เช่น หัวหน้าภาควิชา หมายเลขโทรศัพท์ จำนวนนักศึกษา
- คอมพิวเตอร์ของภาควิชา เช่น หมายเลขประจำเครื่อง ยี่ห้อ ราคา ร้านที่จำหน่าย
- บริษัทที่จำหน่ายคอมพิวเตอร์ เพื่อติดต่อ ซ่อมแซม หรือขอใช้บริการต่าง ๆ เมื่อคอมพิวเตอร์มีปัญหา
- ห้องปฏิบัติการคอมพิวเตอร์ เช่น หมายเลขห้อง อาคาร ผู้ที่รับผิดชอบดูห้อง เวลาเปิด-ปิด

ก) จงเขียนผัง E-R เพื่อออกแบบฐานข้อมูล

ข) จากข้อ ก) จงเขียน Relation schema โดยระบุ PK, FK ให้ชัดเจน

ค) ตรวจสอบ Relation ให้อยู่ใน BCNF



Exercise

9. Mr. Rodonso นักธุรกิจระดับมหาเศรษฐีคนหนึ่งต้องการสร้างฐานข้อมูลส่วนตัวเพื่อเก็บข้อมูลเกี่ยวกับภาพยนตร์ทุกประเภทที่ได้รับรางวัล และถูกนำไปเผยแพร่/เข้าฉายในโรงภาพยนตร์ในประเทศต่าง ๆ ทั่วโลก โดยมีข้อมูลที่ต้องการบันทึกเบื้องต้นดังนี้
- ข้อมูลเกี่ยวกับภาพยนตร์ เช่น ชื่อภาพยนตร์ ปีที่ถ่ายทำ ประเภทภาพยนตร์ (เช่น สายลับ ตลก ชีวิต เป็นต้น) MPAA rating (เช่น RateR, G หรือ PG) รางวัลที่ได้รับ ซึ่งอาจมีหลายรางวัล (เช่น เพลงประกอบยอดเยี่ยม เนื้อหาสร้างสรรค์ เป็นต้น)
 - ข้อมูลผู้อำนวยการสร้าง เช่น ชื่อ วันเกิด สถานะ (เช่น เสียชีวิตแล้ว หรือยังมีชีวิตอยู่) สัญชาติ เป็นต้น
 - ข้อมูลดารานำแสดง เช่น ชื่อและนามสกุลจริง ชื่อและนามสกุลที่ใช้ในการแสดง (ถ้ามี) อายุ เชื้อชาติ สัญชาติ เป็นต้น
 - ข้อมูลเกี่ยวกับรายได้การฉายภาพยนตร์ และระยะเวลาที่เข้าโรงในประเทศต่าง ๆ
- นอกจากนี้ยังต้องการบันทึกด้วยว่า ภาพยนตร์เรื่องใดได้มีการจัดทำในรูปแบบของแผ่นซีดีจำหน่ายแล้วด้วย



Exercise

- จงออกแบบฐานข้อมูลนี้โดยเขียนผังอีอาร์
- จากผังอีอาร์ข้างต้น จงแปลงเป็น Relation ที่ถูกต้อง โดยระบุ PK และ FK (และตรวจสอบว่า Relation ที่ได้อยู่ใน BCNF)



Exercise

10. จงเขียนผังอีอาร์สำหรับฐานข้อมูลของบริษัทขายบ้านและที่ดิน ซึ่งมีกฎเกณฑ์การทำงานพอสรุปได้ดังนี้
- บริษัทมีสำนักงานขาย (Sale office) หลายแห่งในจังหวัดใหญ่ ๆ ทั่วประเทศ ข้อมูลของสำนักงานขายประกอบด้วย รหัสสำนักงาน (office_no) และที่ตั้งสำนักงาน (location)
 - สำนักงานขายแต่ละแห่งมีพนักงานทำงานประจำอย่างน้อย 1 คน โดยพนักงานแต่ละคนประจำสำนักงานขายเพียงแห่งเดียว ข้อมูลของพนักงานอาจประกอบด้วยรหัสพนักงาน (employee_no) ชื่อ (employee_name)
 - สำนักงานขายแต่ละแห่งมีผู้จัดการ 1 คน (ซึ่งเป็นพนักงานของบริษัทด้วย) บริษัทจะแต่งตั้งผู้จัดการ 1 คนให้บริหารสำนักงานเพียงแห่งเดียวเท่านั้น



Exercise

- บริษัทจะจัดทำรายการบ้านหรือที่ดินที่ต้องการขาย (Property for sale) สำหรับแต่ละสำนักงาน โดยที่บ้านหรือที่ดิน 1 แห่งจะจำหน่ายที่สำนักงานขายเพียงแห่งเดียว แต่สำหรับสำนักงานหนึ่งแห่งอาจจำหน่ายบ้าน/ที่ดินหลายแห่ง คุณสมบัติของรายการบ้าน/ที่ดินดังกล่าวประกอบด้วย รหัสบ้าน/ที่ดิน (property_no) ที่อยู่ (address) จังหวัด (city) และรหัสไปรษณีย์ (zip_code)
- สำหรับบ้าน/ที่ดินที่จะจำหน่ายนี้จะระบุเจ้าของ (owner) ด้วย โดยเจ้าของบ้าน/ที่ดินแต่ละแห่งอาจมีมากกว่า 1 คนก็ได้ คุณสมบัติของเจ้าของประกอบด้วยรหัสเจ้าของ (owner_id) และชื่อเจ้าของ (owner_name)

นักศึกษาอาจตั้งสมมติฐานหรือเพิ่มเติม attribute บางอย่างตามความจำเป็น



Exercise

11. **Project, Inc.** เป็นโรงงานที่มีพนักงานเกือบ ๆ 500 คน ต้องการสร้างฐานข้อมูลเพื่อเก็บข้อมูลที่เกี่ยวข้องกับตำแหน่ง ทักษะของพนักงาน ข้อมูลโครงการที่พนักงานแต่ละคนถูกกำหนดให้ทำ และแผนกที่พนักงานทำงาน

พนักงานแต่ละคนจะมีรหัสที่โรงงานออกให้โดยไม่ซ้ำกันเลย ถ้าพนักงานคนใดสมัครกับพนักงานคนอื่นในโรงงานเดียวกันนี้ ฐานข้อมูลก็จะบันทึกข้อมูลนี้พร้อมกับวันที่สมัคร และฐานข้อมูลนี้ต้องการเก็บข้อมูลที่เป็นปัจจุบันเท่านั้น

โรงงานนี้แบ่งเป็น 11 แผนก พนักงานแต่ละคนจะทำงานในแผนกเดียวเท่านั้น และถ้าแต่ละแผนกต้องการซื้ออุปกรณ์ต่าง ๆ ก็จะติดต่อโดยตรงกับผู้ขาย ปกติผู้ขายรายหนึ่ง ๆ ก็จะขายอุปกรณ์ให้หลายแผนก ฐานข้อมูลนี้จึงต้องการเก็บบันทึกข้อมูลชื่อและที่อยู่ของผู้ขายเหล่านี้ด้วย

สำหรับแต่ละโครงการของโรงงาน มักจะมีพนักงานหลายคนทำงานบนโครงการเดียวกัน และพนักงานบางคนได้รับมอบหมายให้ทำงานหลายโครงการ แต่พนักงานหนึ่งคนทำโครงการในจังหวัดหนึ่งได้เพียงโครงการเดียวเท่านั้น (คือต้องไม่ทำหลายโครงการในจังหวัดเดียวกัน) ข้อมูลจังหวัดที่ต้องการ คือ ชื่อจังหวัด และประชากรของจังหวัดนั้น



Exercise

พนักงานคนหนึ่ง ๆ ของบริษัทอาจมีความสามารถพิเศษหลายด้าน โดยการทำงานโครงการของพนักงาน จำเป็นต้องใช้ความสามารถพิเศษ ซึ่งพนักงานใช้ความสามารถพิเศษที่เขามีอยู่อย่างน้อยหนึ่งด้านในโครงการที่ได้รับมอบหมายเสมอ (สำหรับความสามารถพิเศษนี้ สิ่งที่ต้องการบันทึกในฐานข้อมูลคือ คำอธิบายทักษะ และหมายเลขหรือรหัสที่กำหนดให้สำหรับความสามารถพิเศษ)

- 1) จงเขียนแผนภาพอีอาร์สำหรับบริษัทนี้
- 2) จากข้อ 1) จงเขียน Relation ที่สอดคล้อง พร้อมระบุ PK และ FK ให้ถูกต้อง



Exercise

12. บริษัท Notown Records ต้องการสร้างฐานข้อมูลเพื่อเก็บข้อมูลของนักดนตรี (musicians) ที่ออกอัลบั้มของตัวเอง และข้อมูลอื่น ๆ ที่เกี่ยวข้องด้วย บริษัทจึงตัดสินใจว่าจ้างท่านให้ออกแบบฐานข้อมูลของบริษัท
- ข้อมูลนักดนตรีแต่ละคนประกอบด้วย หมายเลขประจำตัวปชช. (SSN) ชื่อ ที่อยู่ หมายเลขโทรศัพท์ มีนักดนตรีบางคนอาศัยอยู่ร่วมกับนักดนตรีคนอื่น แต่ไม่มีนักดนตรีคนใดใช้หมายเลขโทรศัพท์มากกว่า 1 หมายเลข
 - ข้อมูลเครื่องดนตรีแต่ละชิ้นที่ใช้ในการเล่นประกอบเพลง ประกอบด้วยชื่ออุปกรณ์ เช่น กีตาร์ ขลุ่ย และ musical key เช่น C B-flat E-flat เป็นต้น
 - ข้อมูลของแต่ละอัลบั้มประกอบด้วย ชื่ออัลบั้ม วันที่จดลิขสิทธิ์ (copyright date) และ รหัสอัลบั้ม ข้อมูลแต่ละเพลงซึ่งประกอบด้วยชื่อเพลงและผู้แต่งเพลง
 - นักดนตรีแต่ละคนอาจจะเล่นเครื่องดนตรีหลายประเภท และเครื่องดนตรีประเภทหนึ่ง มีนักดนตรีหลายคนที่สามารถเล่นได้



Exercise

- แต่ละอัลบั้มมีหลายเพลง แต่เพลงหนึ่ง ๆ จะอยู่ในอัลบั้มเดียวเท่านั้น
 - แต่ละอัลบั้มมีนักดนตรีเพียงคนเดียวที่เป็น **producer** และนักดนตรีคนหนึ่งสามารถมีหลายอัลบั้มได้
- 1) จงเขียนผังอีอาร์สำหรับข้อมูล โดยระบุ **Identifier (key)** และ **Cardinality constraints** นักศึกษาสามารถกำหนดสมมติฐานเพิ่มเติมได้
 - 2) จากผังอีอาร์จงเขียน **Relational schema** ให้ถูกต้อง



Exercise

13. สมมติท่านและทีมงานได้รับมอบหมายให้ออกแบบฐานข้อมูลของโรงพยาบาลเอกชนแห่งหนึ่ง ทีมงานได้สัมภาษณ์ผู้บริหารและพนักงานหลายคนของโรงพยาบาลเพื่อศึกษาวิธีการทำงาน และได้ระบุเอนทิตีหลักสำหรับโรงพยาบาล พบว่า

- ในรพ.นี้มีตึกคนไข้ (ward) หลายตึก เช่น ตึกเด็ก ตึกคนไข้ชาย ตึกคนไข้หญิง แต่ละตึกมีชื่อเรียกต่าง ๆ กันและไม่ซ้ำกัน
- คนไข้ของรพ.ประกอบด้วยคนไข้ใน (resident) และคนไข้นอก (outpatient) คนไข้ในต้องมีการบันทึกข้อมูลวันที่เข้ารพ. (data-admitted) ด้วย
- แพทย์รับผิดชอบในการรักษาคนไข้ ซึ่งบางครั้งคนไข้ 1 คนอาจถูกรักษาโดยแพทย์มากกว่า 1 คนก็ได้
- ยาหรืออุปกรณ์ที่ใช้ในการรักษาคนไข้ สำหรับการรักษาแต่ละครั้งจะเก็บเงินจากคนไข้ทั้งหมด



Exercise

- ทีมงานยังได้พบอีกว่า รพ.นี้มีบุคคลที่เกี่ยวข้อง 4 กลุ่ม คือ
 - ก. พนักงาน (employee) ต้องบันทึกข้อมูลวันที่เริ่มทำ (data-hired)
 - ข. แพทย์ (physician) ต้องบันทึกข้อมูลความเชี่ยวชาญ (specialty) และ pager number
 - ค. คนไข้ (patient) ต้องบันทึกข้อมูลวันที่มาติดต่อกับรพ. (contact-date)
 - ง. อาสาสมัคร (volunteer) ต้องบันทึกข้อมูลทักษะพิเศษ (skill)
- ทีมงานพบว่ามีคนอีกกลุ่มหนึ่ง (ส่วนน้อยมาก) ไม่อยู่ในทั้ง 4 กลุ่มข้างต้น และบางคนอาจอยู่มากกว่า 1 กลุ่ม เช่น อาสาสมัครบางคนเป็นคนไข้ของรพ.ด้วย
- คนไข้แต่ละคนของรพ.จะมีแพทย์ 1 คนเป็นเจ้าของไข้ ในบางเวลาแพทย์บางคนอาจมีคนไข้มากกว่า 1 คนที่ต้องรับผิดชอบหรืออาจไม่มีเลยก็ได้



Exercise

- พนักงาน (employee) ในรพ.แบ่งเป็น 3 กลุ่ม
 - ก. กลุ่มพยาบาล (nurse)
 - ข. กลุ่มผู้ปฏิบัติงานในห้องปฏิบัติการ (technician) และ
 - ค. พนักงานทั่วไป (staff) เช่น พนักงานธุรการ การเงิน เป็นต้น
- กลุ่มพยาบาลมีลักษณะเฉพาะ (attribute) คือ คุณวุฒิ (qualification) พนักงานทั่วไปมีข้อมูลเฉพาะเกี่ยวกับระดับชั้นงาน (job-class) และผู้ปฏิบัติงานในห้องปฏิบัติการต้องมีข้อมูลเกี่ยวกับงานเฉพาะด้านที่ทำ (job-type) นอกจากนี้พยาบาลแต่ละคนจะถูกกำหนดให้ทำงานประจำในแต่ละตึกคนไข้ โดยแต่ละตึกคนไข้อาจมีพยาบาลหลายคนก็ได้ และแต่ละตึกก็มีหัวหน้าพยาบาล 1 คนทำหน้าที่เป็นหัวหน้าตึกคนไข้นั้น ๆ
- ส่วนผู้ปฏิบัติงานในห้องปฏิบัติการจะถูกกำหนดให้ทำงานในห้องปฏิบัติการหนึ่งห้องหรือมากกว่า (ห้องปฏิบัติการมีลักษณะเฉพาะ คือ ชื่อห้องปฏิบัติการ และตำแหน่งที่ตั้ง) ดังนั้นในแต่ละห้องปฏิบัติการจึงต้องมีผู้ปฏิบัติงานอย่างน้อย 1 คนเสมอ



Exercise

- คนใช้ที่นอนรพ. ต้องมีการกำหนดเตียงคนไข้ (bed) ให้เสมอ เตียงคนไข้นี้อ้างถึงโดยใช้ bed_id (ซึ่งประกอบด้วยค่าของ bed# และ room#)
- 1) จงเขียนผังอีอาร์ทีที่สอดคล้องกับความต้องการข้างต้น
 - 2) สร้าง Relational schema จากผังที่ได้ในข้อ 1 ให้อยู่ในรูป 3NF



Exercise

14. มหาวิทยาลัยแห่งหนึ่งมีหน่วยงานดูแลสวัสดิการด้านที่พักให้แก่บุคลากร โดยมีรายละเอียดต่าง ๆ ดังนี้
- ข้อมูลบุคลากรเก็บชื่อและนามสกุล เลขประจำตัวปชช. (SSN) คณะที่สังกัด ที่อยู่และเบอร์โทรศัพท์ที่ติดต่อได้ สถานะการสมรส (โสด แต่งงาน กรณีที่หย่าให้เก็บสถานะเป็นโสด)
 - ที่พักที่ให้บริการมีหลายประเภท ได้แก่ หอพัก (Dormitory) อพาร์ทเมนต์ (Apartment) บ้านเดี่ยว (House)
 - กรณีที่เป็นหอพักและอพาร์ทเมนต์จะมีชื่ออาคาร หมายเลขห้อง และลักษณะของห้อง กรณีที่เป็นบ้านเดี่ยวจะมีหมายเลขบ้านและลักษณะบ้าน
 - บุคลากร 1 คนจะขอที่พักได้แห่งเดียวเท่านั้น บุคลากรมหาวิทยาลัยที่แต่งงานกันมีสิทธิ์ขอหอพักได้เพียงแห่งเดียวเช่นเดียวกัน
 - มีการเก็บประวัติการเข้าพักของบุคลากรในที่พักต่าง ๆ ได้แก่ วันที่เข้าพัก วันที่คืนที่พัก