

แบบจำลองเอนทิตี (Entity Relationship Model)

ความรู้พื้นฐาน (Basic Concepts)

- ไดอะแกรมอีอาร์ (ER-Diagram:ERD) เป็นแผนภาพที่แสดงรายละเอียดรวมถึงความสัมพันธ์ระหว่างข้อมูล
- โดยใช้แผนภาพในการนำเสนอ เพื่อช่วยให้ง่ายต่อการทำความเข้าใจ
- และเพื่อใช้เป็นเครื่องมือในการสื่อสารกับบุคลากรต่างๆ ที่เกี่ยวข้องกัระบบฐานข้อมูล เช่น นักออกแบบฐานข้อมูล โปรแกรมเมอร์ ผู้ใช้งาน ฯลฯ
- ไดอะแกรมอีอาร์ ประกอบด้วยส่วนประกอบที่สำคัญ คือ
 - เอนทิตี (Entity)
 - แอตทริบิวต์ (Attributes)
 - ความสัมพันธ์ (Relationship)



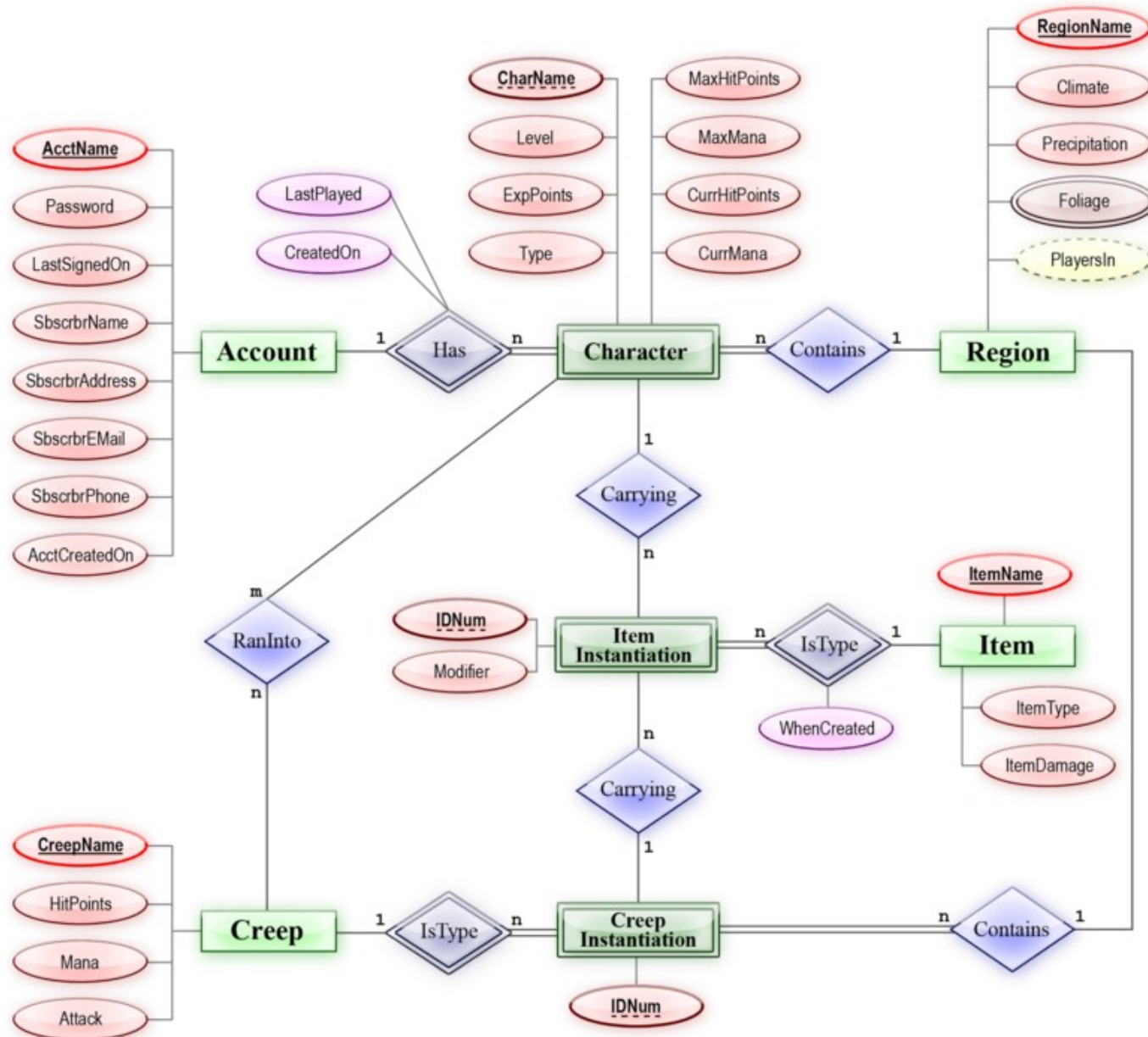
การสร้างโดยโปรแกรมอีอาร์

Chen's Model

ประกอบด้วยสัญลักษณ์พื้นฐานดังนี้

- เอนทิตี แทนด้วย รูปสี่เหลี่ยม
- แอตทริบิวต์ แทนด้วย รูปวงรี
- ความสัมพันธ์ แทนด้วย รูปสี่เหลี่ยมขนมเปียกปูน
- เส้นตรง (Line) ที่ลากเชื่อมระหว่าง เอนทิตีและแอตทริบิวต์ หมายถึง แอตทริบิวต์ เป็นของเอนทิตีนั้นๆ และใช้ลากเพื่อแสดงความสัมพันธ์ระหว่างเอนทิตี
- ขีดเส้นใต้ (Underline) ใช้เพื่อแสดงว่า แอตทริบิวต์นั้นเป็นคีย์หลักของเอนทิตี
- ภายในสัญลักษณ์จะต้องมีตัวระบุ หรือชื่อ และควรเป็นชื่อที่สื่อถึงเอนทิตี แอตทริบิวต์ หรือ ความสัมพันธ์นั้นๆ

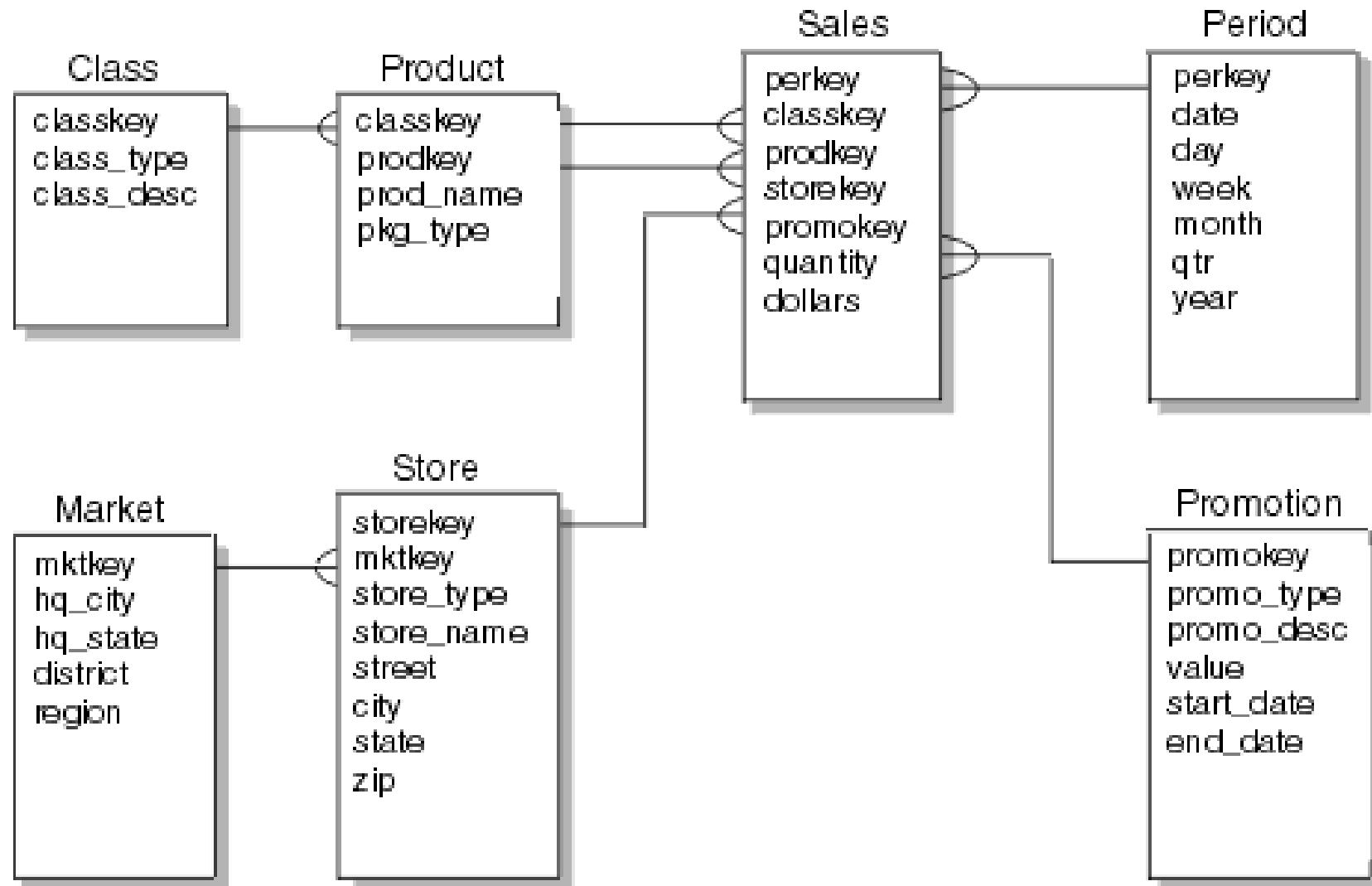
Chen's Model



Crow's Foot Model

- ใช้สีเหลี่ยมแทนเอนทิตี
- แอตทริบิวต์ของเอนทิตี ใช้การใส่ตัวอักษรลงภายในเอนทิตี โดยมีเส้นตรงขีดชั้นระหว่างชื่อของเอนทิตีและแอตทริบิวต์
- ใช้เส้นตรงแสดงความสัมพันธ์ระหว่างเอนทิตีเช่นเดียวกัน

Crow's Foot Model



เอนทิตี (Entity)

- หมายถึง สิ่งที่เราสนใจ อาจเป็นคน สัตว์ สิ่งของ เหตุการณ์โดยที่เอนทิตีเหล่านี้มีคุณสมบัติร่วมกัน เช่น เอนทิตี้นักศึกษา เอนทิตี้อาจารย์ เอนทิตีห้องเรียน ฯลฯ
- เอนทิตีสามารถแบ่งเป็นประเภทต่างๆ ได้ดังนี้
 - **Strong Entity** หมายถึง เอนทิตีที่เกิดขึ้นได้ด้วยตัวเองโดยไม่ขึ้นกับเอนทิตีใดๆ
 - **Weak Entity** หมายถึง เอนทิตีที่เกิดหรือการคงอยู่ (Exist) ขึ้นกับเอนทิตีอื่นๆ

Strong Entity

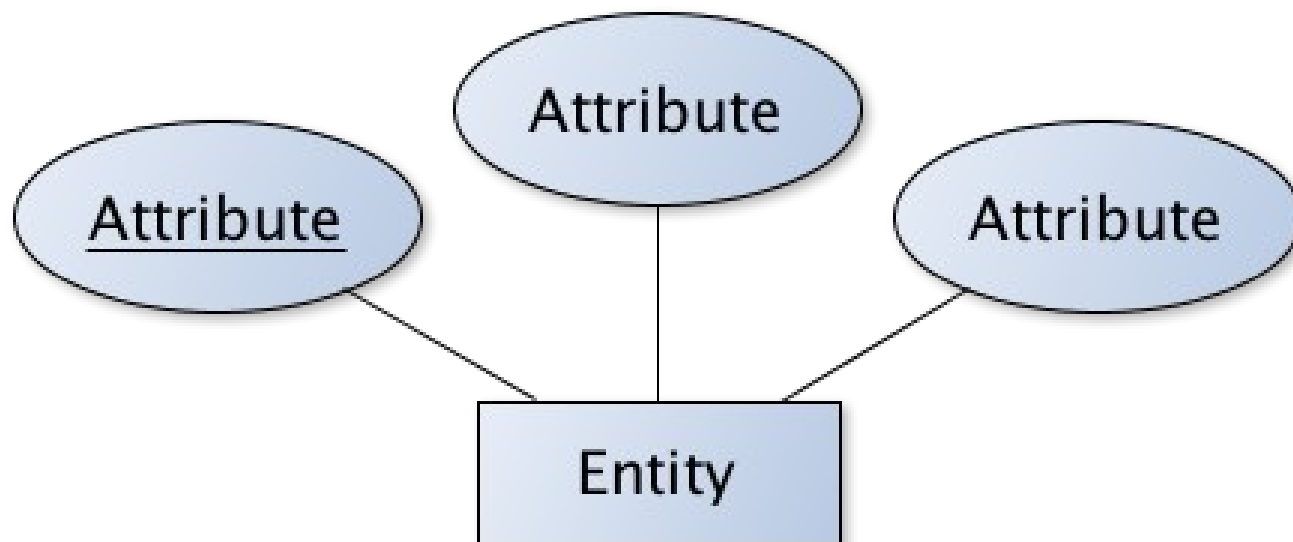
Weak Entity

แอตทริบิวต์ (Attributes)

- หมายถึง คุณสมบัติของเอนทิตี หรือคุณลักษณะที่ใช้ในการอธิบายเอนทิตีนั้นๆ
- แอตทริบิวต์สามารถแบ่งเป็นประเภทต่างๆ ได้ดังนี้
 - Simple Attribute
 - Composite Attribute
 - Single Value Attribute
 - Multi-value Attribute
 - Derived Attribute

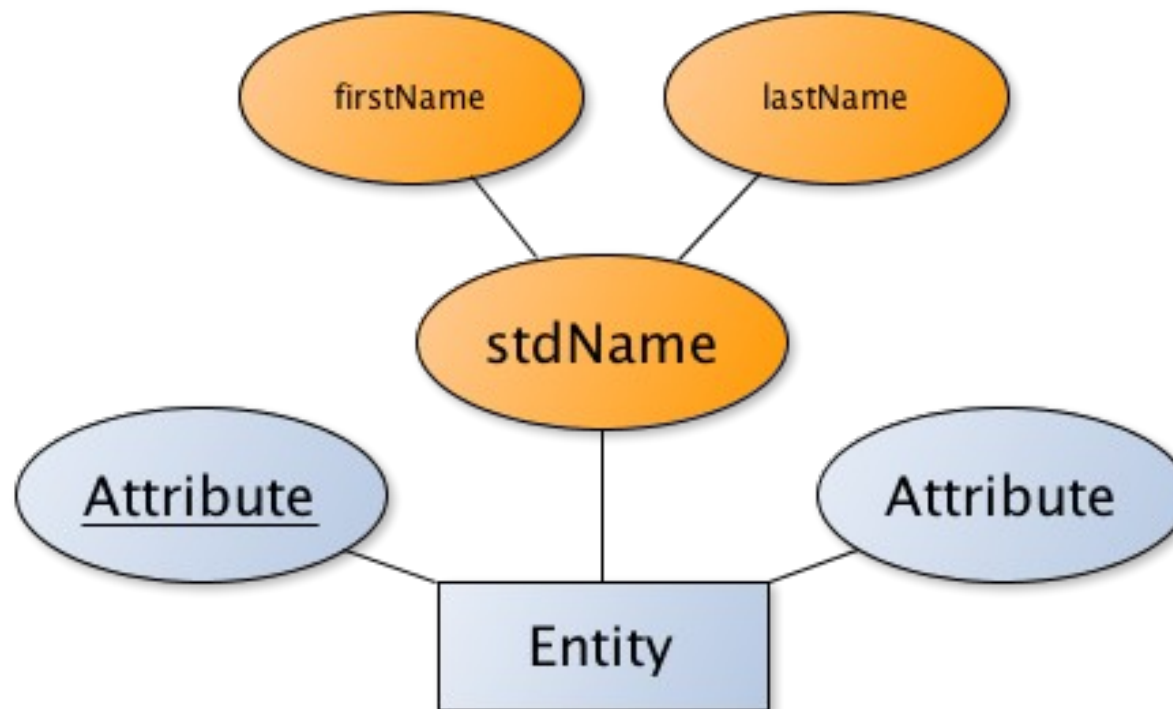
Simple Attribute

- หมายถึง แอตทริบิวต์ที่มีค่าเดียว ไม่สามารถแบ่งย่อยได้อีก



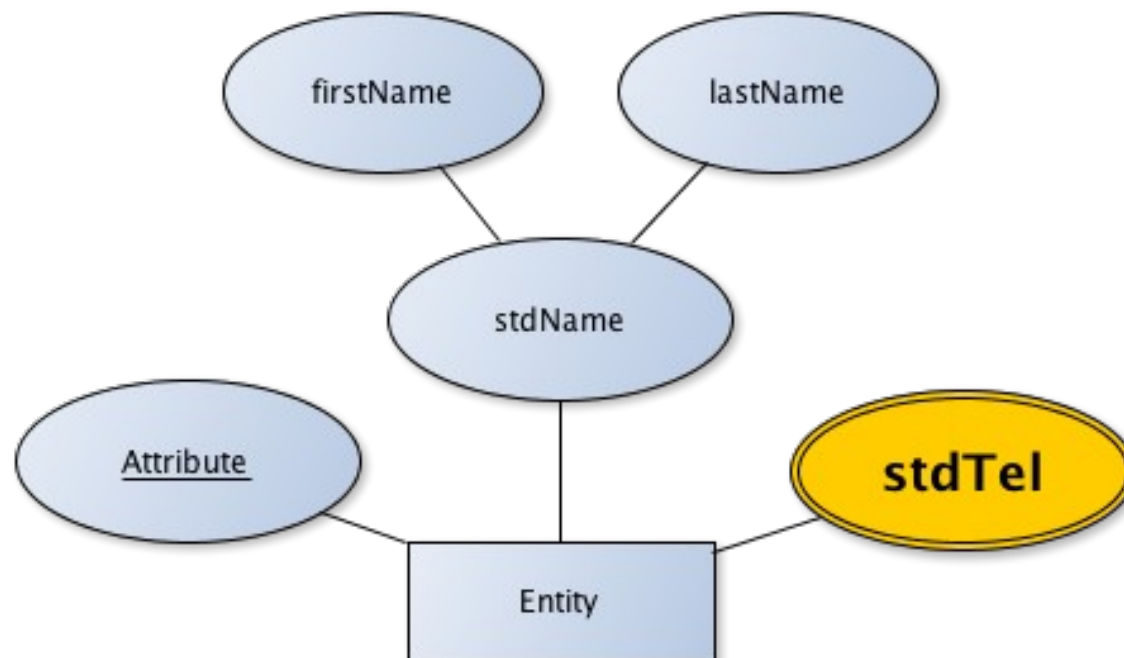
Composite Attribute

- หมายถึง แอตทริบิวต์ที่ประกอบไปด้วยแอตทริบิวต์ย่อย เช่น stdName ประกอบไปด้วย first-name และ last-name



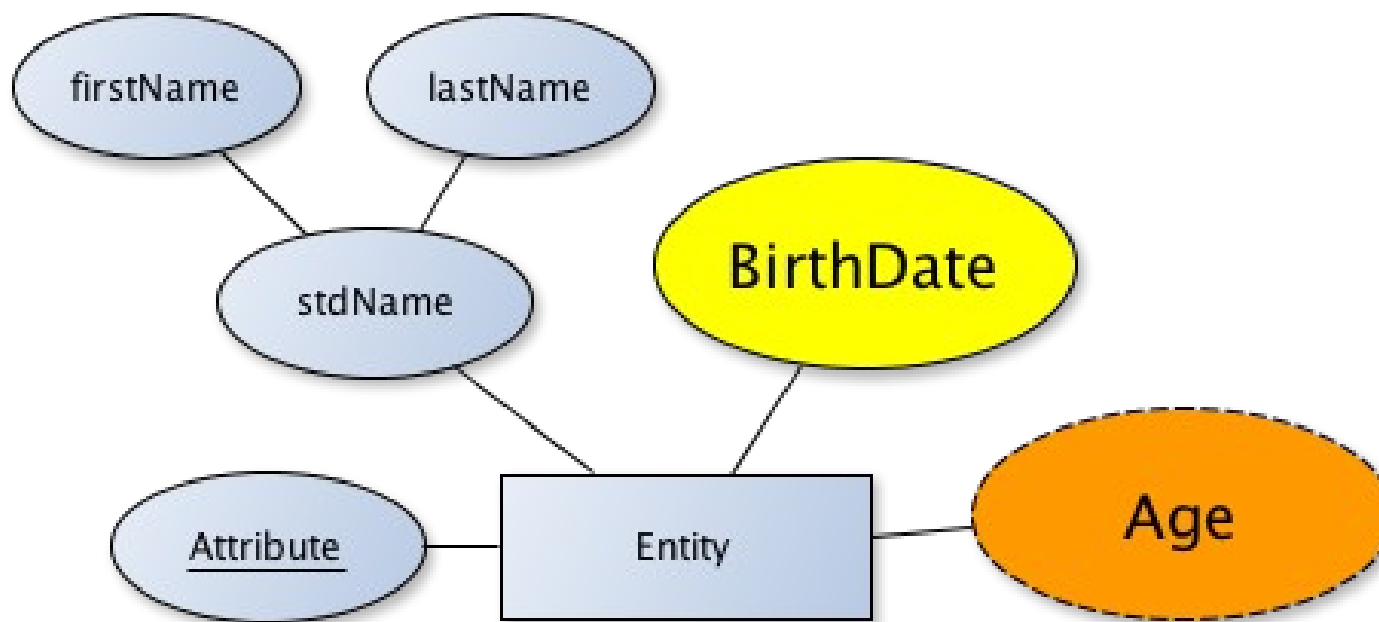
Single Value and Multi-value Attribute

- **Single Value** หมายถึง แอตทริบิวต์ที่บรรจุค่าเพียงค่าเดียว
- **Multi-value Attribute** หมายถึง แอตทริบิวต์ที่บรรจุค่าได้หลายค่า เช่น
แอตทริบิวต์ stdTel ที่ใช้เก็บหมายเลขโทรศัพท์ของนักศึกษา ซึ่งหนึ่งคนอาจมีได้
หลายหมายเลข เป็นต้น



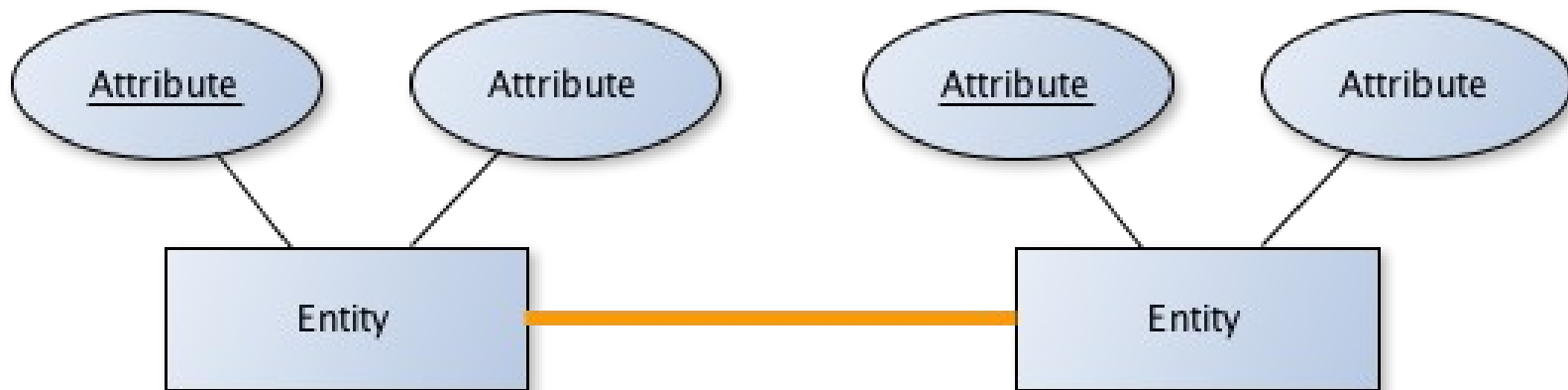
Derived Attribute

- หมายถึง แอตทริบิวต์ที่มีค่าแปรเปลี่ยนไปตามขนาดของค่าแอตทริบิวต์อื่น เช่น stdAge ขึ้นต่อ stdDOB



ความสัมพันธ์ (Relationship)

- หมายถึง ความสัมพันธ์ระหว่างเอนทิตี กล่าวคือ แต่ละรายการข้อมูล (Entity Instance) ของเอนทิตีหนึ่ง มีความสัมพันธ์กับรายการข้อมูลของเอนทิตีอื่นๆ



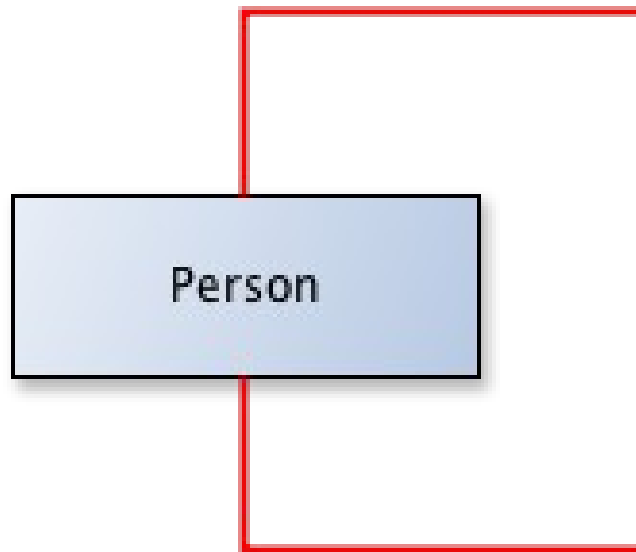
ทฤษฎีของความสัมพันธ์

คือ จำนวนเอนทิตีที่มีส่วนร่วมของความสัมพันธ์ระหว่างกัน แบ่งออกเป็นประเภทต่างๆ ได้ดังนี้

- ความสัมพันธ์เอกภาค (Unary Relationship)
- ความสัมพันธ์ทวิภาค (Binary Relationship)
- ความสัมพันธ์ไตรภาค (Ternary Relationship)

ความสัมพันธ์เอกภาค (Unary Relationship)

- ความสัมพันธ์ระหว่าง 1 เอนทิตี



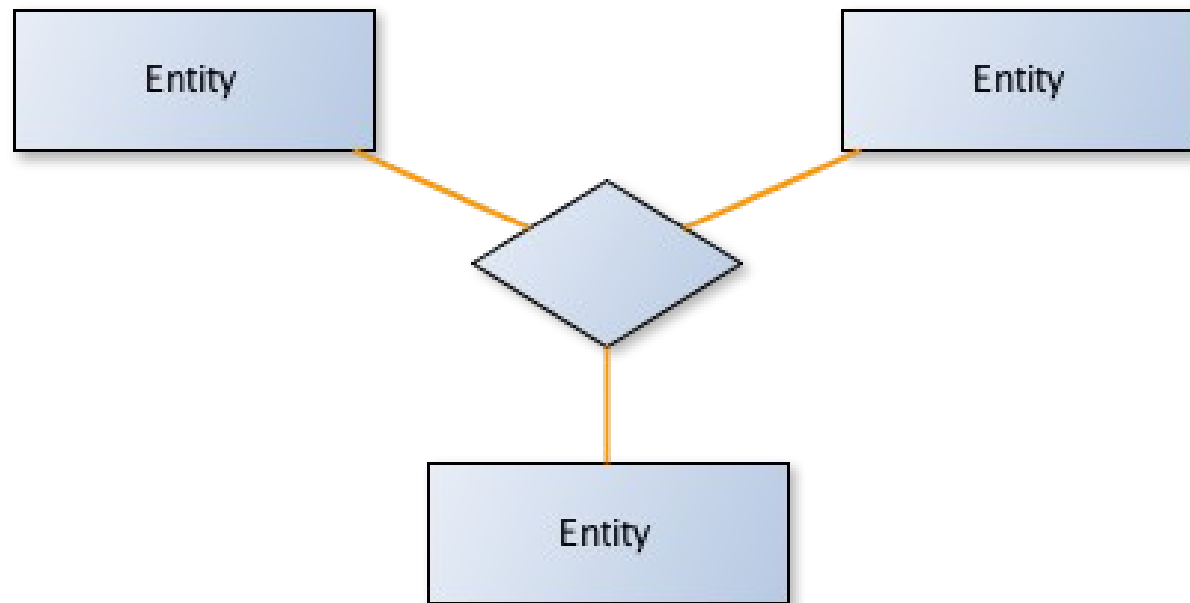
ความสัมพันธ์ทวิภาค (Binary Relationship)

- ความสัมพันธ์ระหว่าง 2 เอนทิตี



ความสัมพันธ์ไตรภาค (Ternary Relationship)

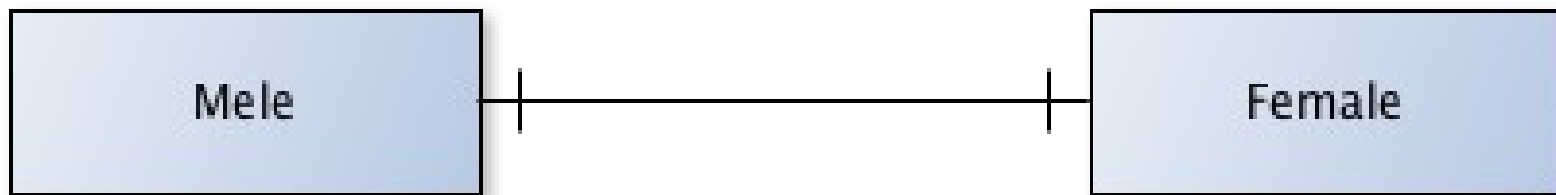
- ความสัมพันธ์ระหว่าง 3 เอนทิตี



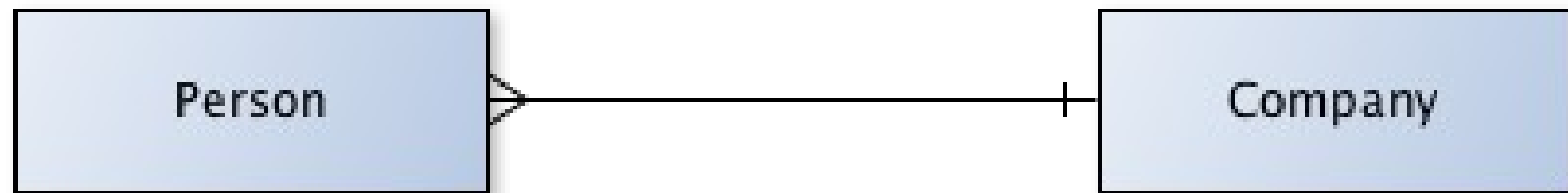
Cardinality of Relationship

- หมายถึงจำนวนรายการข้อมูลในแต่ละเอนทิตี (Entity Instance) ที่มีความสัมพันธ์กับแต่ละรายการข้อมูลของเอนทิตีอื่น ๆ
- แบ่งออกเป็น
 - ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One-to-One Relationship [1:1])
 - ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One-to-Many Relationship [1:M])
 - ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many-to-Many Relationship [M:M | M:N])

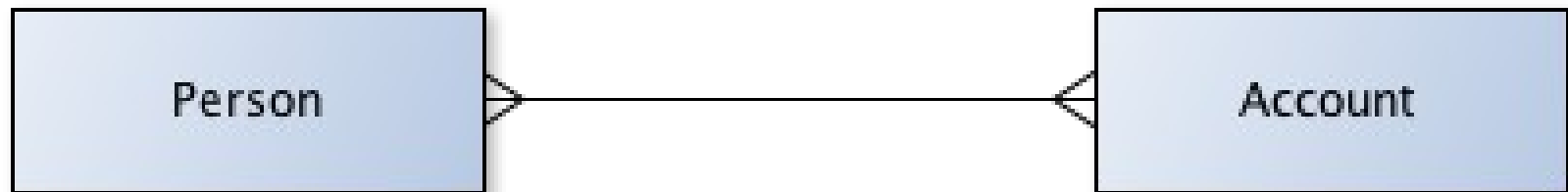
ความสัมพันธ์แบบหนึ่งต่อหนึ่ง
(One-to-One Relationship [1:1])



ความสัมพันธ์แบบหนึ่งต่อกลุ่ม
(One-to-Many Relationship [1:M])



ความสัมพันธ์แบบกลุ่มต่อกลุ่ม
(Many-to-Many Relationship [M:M | M:N])



กฎเกณฑ์ข้อกำหนดของความสัมพันธ์ (Constraints)

- ข้อกำหนดของความสัมพันธ์ เป็นกฎเกณฑ์ที่ใช้เป็นข้อบังคับ เงื่อนไข เพื่อให้การจัดเก็บข้อมูลในฐานข้อมูลเป็นไปอย่างเหมาะสมและมีความถูกต้อง
- ข้อกำหนดจะเป็นเงื่อนไขที่ใช้บังคับส่วนต่างๆ ในแบบจำลองซึ่งโปรแกรมจะต้องรักษาให้ถูกต้องตามความเป็นจริงเสมอ
- ข้อกำหนดดังกล่าวแบ่งเป็นสองชนิดคือ
 - Participation Constraints
 - Cardinality Constraints

Participation Constraints

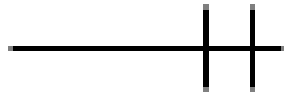
- มีกฎเกณฑ์อยู่สองชนิดคือ
 - **Total** หมายถึง เป็นการบังคับว่าจะต้องมีความสัมพันธ์เกิดขึ้น
 - **Partial** หมายถึง ความสัมพันธ์นั้นจะมีหรือไม่มีก็ได้



Cardinality Constraints

- หมายถึง จำนวนรายการข้อมูลของแต่ละเอนทิตีที่มีความสัมพันธ์ระหว่างกัน โดยแสดงให้เห็นจำนวนที่น้อยที่สุด และจำนวนที่มากที่สุด
- **Maximum Cardinality** หมายถึง จำนวนที่มากที่สุดที่แต่ละรายการข้อมูลของเอนทิตีหนึ่งมีความสัมพันธ์กับเอนทิตีอื่น
- **Minimum Cardinality** หมายถึง จำนวนที่น้อยที่สุดที่แต่ละรายการข้อมูลของเอนทิตีหนึ่งมีความสัมพันธ์กับเอนทิตีอื่น

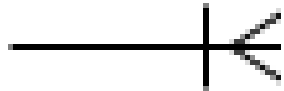
Cardinality Constraints



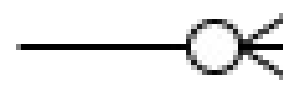
Mandatory One (min = 1, max = 1)



Optional One (min = 0, max = 1)



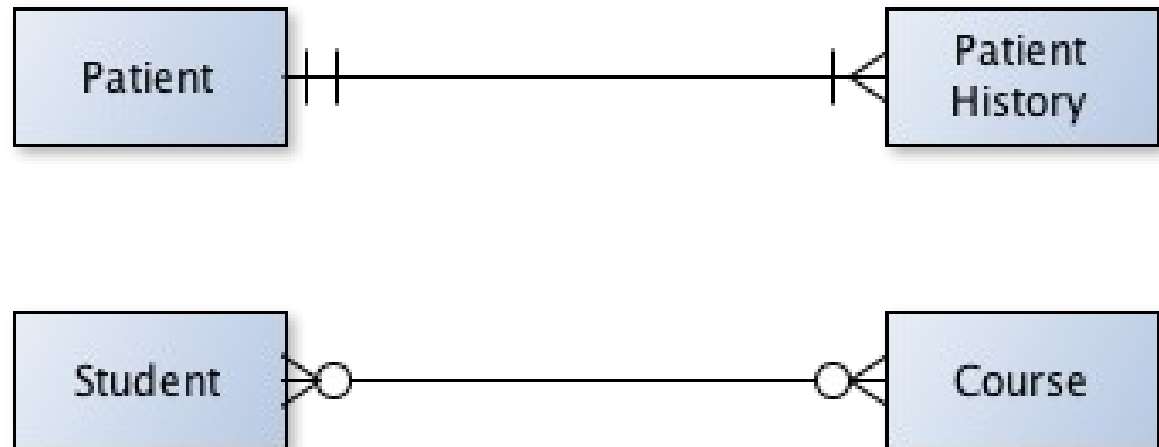
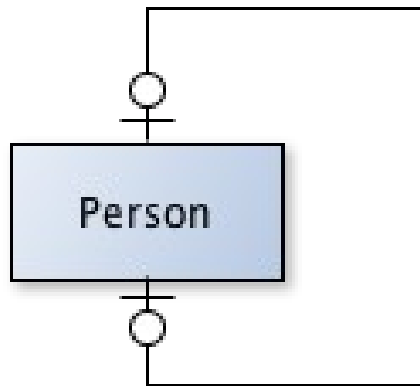
Mandatory Many (min = 1, max = 1)



Optional Many (min = 0, max = 1)

Cardinality Constraints

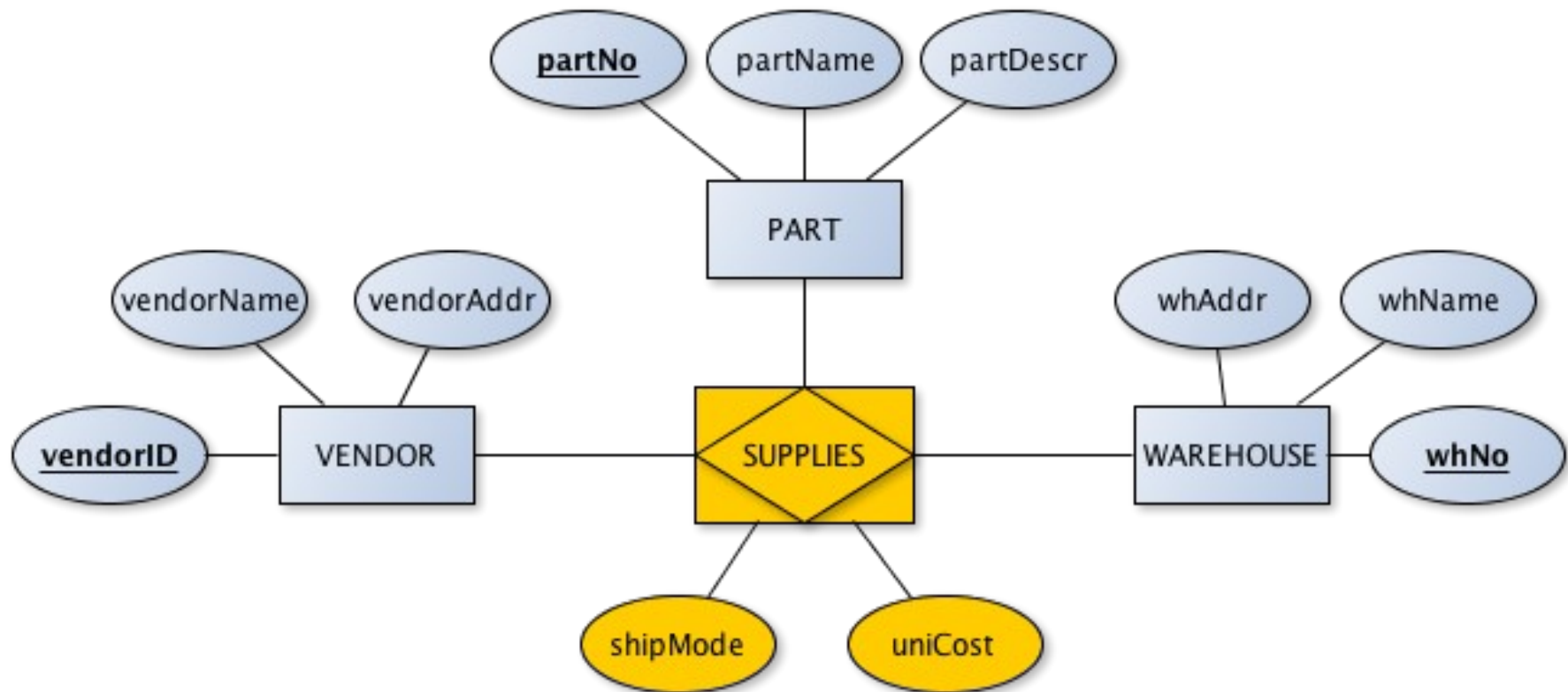
- ตัวอย่าง



Example

- ไดอะแกรมอีอาร์แสดงความสัมพันธ์ระหว่างพนักงาน (VENDOR) สินค้า (PART) และโรงงาน (WAREHOUSE) เมื่อพนักงานแต่ละคนสามารถขายสินค้าหลายชนิดให้หลายโรงงาน แต่ละโรงงานอาจจะซื้อสินสินค้าหลายชนิดจากพนักงานแต่ละคน และสินค้าแต่ละชนิดถูกขายโดยพนักงานหลายคนเพื่อส่งให้โรงงานหลายแห่ง โดยให้ระบุวิธีการส่งสินค้า (Shipping Mode) และค่าใช้จ่ายในการขนส่งด้วย (Unit Cost) ด้วย

Associative Entity



แบบจำลองอีอีอาร์ (Enhanced ER Model : EERM)

- ปัจจุบันระบบงานต่างๆ ถูกพัฒนาตามเทคโนโลยี ทำให้มีความซับซ้อนมากขึ้น จึงมีการพัฒนาแบบจำลองเพื่อให้สามารถนำเสนอระบบที่ซับซ้อนได้ดีขึ้น
- นั่นคือแบบจำลอง EERM และนำเสนอด้วยไดอะแกรมอีอีอาร์ (Enhance ER Diagram : EERD)

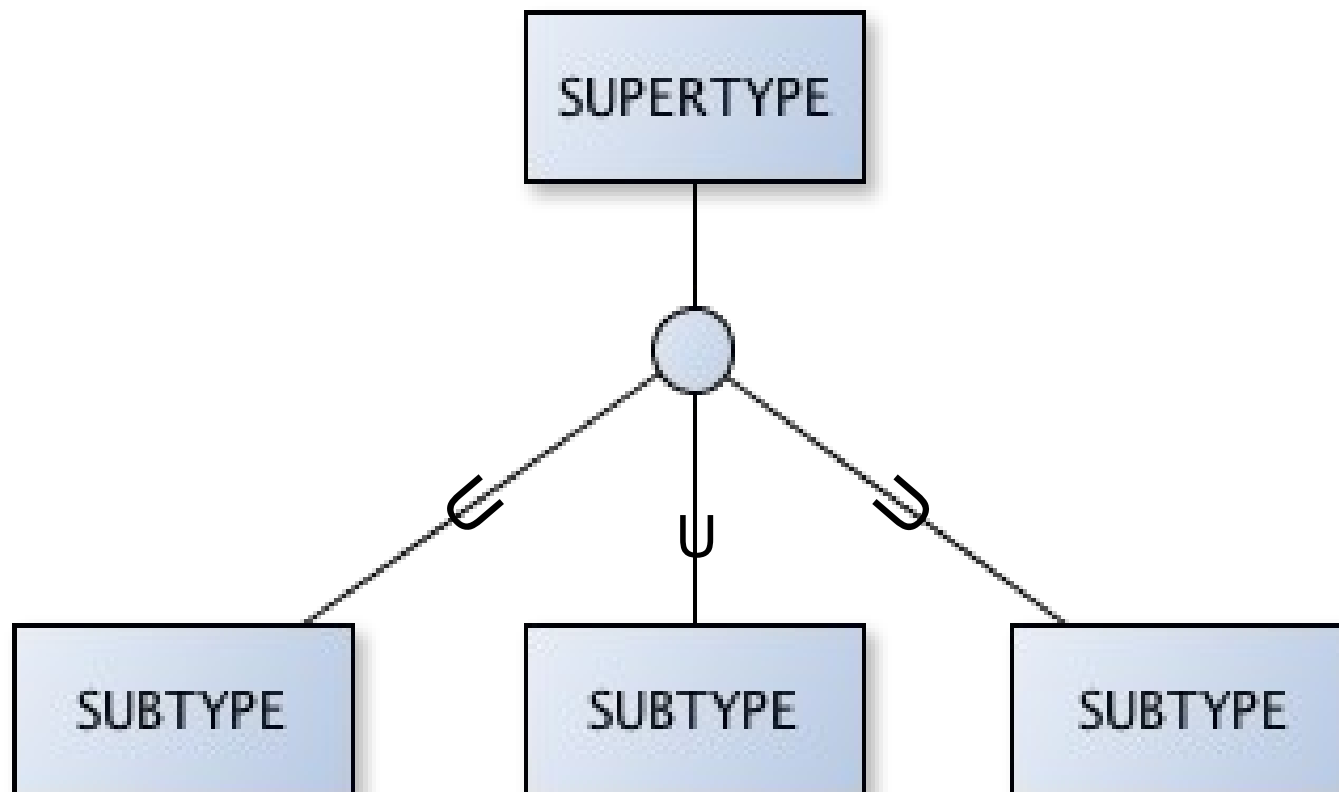
Supertype และ Subtype

- Supertype คือรูปแบบของเอนทิตีที่เป็นต้นแบบของเอนทิตีอื่นๆ โดย supertype จะประกอบไปด้วย subtype ต่างๆ
- Subtype คือเอนทิตีที่มีคุณสมบัติที่แตกต่างกันในกลุ่มของสมาชิก subtype ตัวอื่นๆ
- ตัวอย่างเช่น เอนทิตี staff ที่ประกอบไปด้วยสมาชิกต่างๆ ไม่ว่าจะเป็น manager, secretary และ sales

การถ่ายทอดคุณสมบัติ (Attribute Inheritance)

- Subtype คือเอนทิตีชนิดหนึ่ง ที่ได้รับการถ่ายทอดคุณสมบัติทุกๆ อย่างจาก Supertype
- เช่น Supertype ของรถยนต์ จะประกอบด้วย Subtype ต่างๆ เช่น รถกระบะ รถเก๋ง รถขับเคลื่อน 4 ล้อ เป็นต้น
- ในการออกแบบโมเดลที่มีความสัมพันธ์แบบ Supertype และ Subtype ได้มีกระบวนการเพื่อนำมาจัดการความสัมพันธ์ดังกล่าวอยู่สองกระบวนการคือ
 - **Generalization** เป็นกระบวนการจัดการเอนทิตีที่เป็นแม่แบบเพื่อใช้กำหนดลักษณะเพื่อใช้งานร่วมกัน คือมองหาสิ่งที่เหมือนกันใน Subtype ก่อน
 - **Specialization** เป็นกระบวนการจัดการเอนทิตีที่มีความแตกต่างกันในสมาชิกของกลุ่ม คือมองจุดที่แตกต่างกันระหว่างเอนทิตี หรือต้องกำหนด Supertype และเอนทิตีทั้งหมดออกมาก่อน

แนวคิดพื้นฐานและสัญลักษณ์



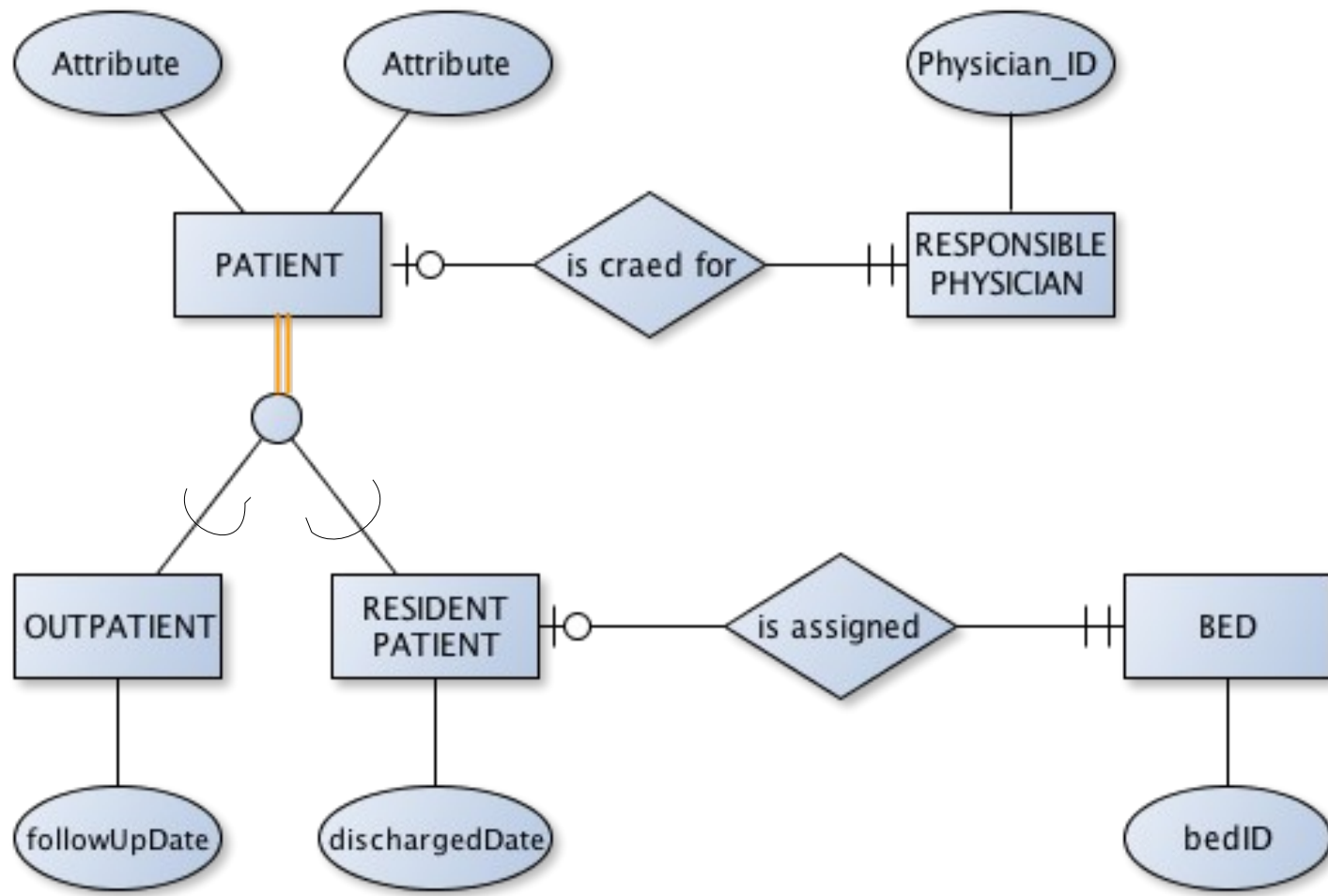
ข้อกำหนด (Constraints) ในความสัมพันธ์ของ Specialization และ Generalization

- เพื่อให้เกิดความถูกต้องและเป็นไปตาม Business Rule
- มีอยู่ 2 ชนิด คือ
 - Completeness Constraints
 - Disjointness Constraints

Completeness Constraints

- เป็นข้อกำหนดที่ใช้พิจารณาว่ารายการตัวอย่าง (Instance) ที่ปรากฏในเอนทิตีที่เป็น Supertype จะต้องปรากฏในเอนทิตีที่เป็น Subtype ด้วยหรือไม่
- แบ่งออกเป็น
 - **Total Specialization Rule** เป็นกฎเกณฑ์ที่กำหนดว่าเอนทิตีที่เป็น Supertype จะ ต้อง มีความสัมพันธ์กับ Subtype ใด Subtype หนึ่ง
 - **Partial Specialization Rule** เป็นกฎเกณฑ์ที่กำหนดว่าเอนทิตีที่เป็น Supertype อาจจะมีหรือไม่มีความสัมพันธ์ใดๆ กับ subtype ก็ได้

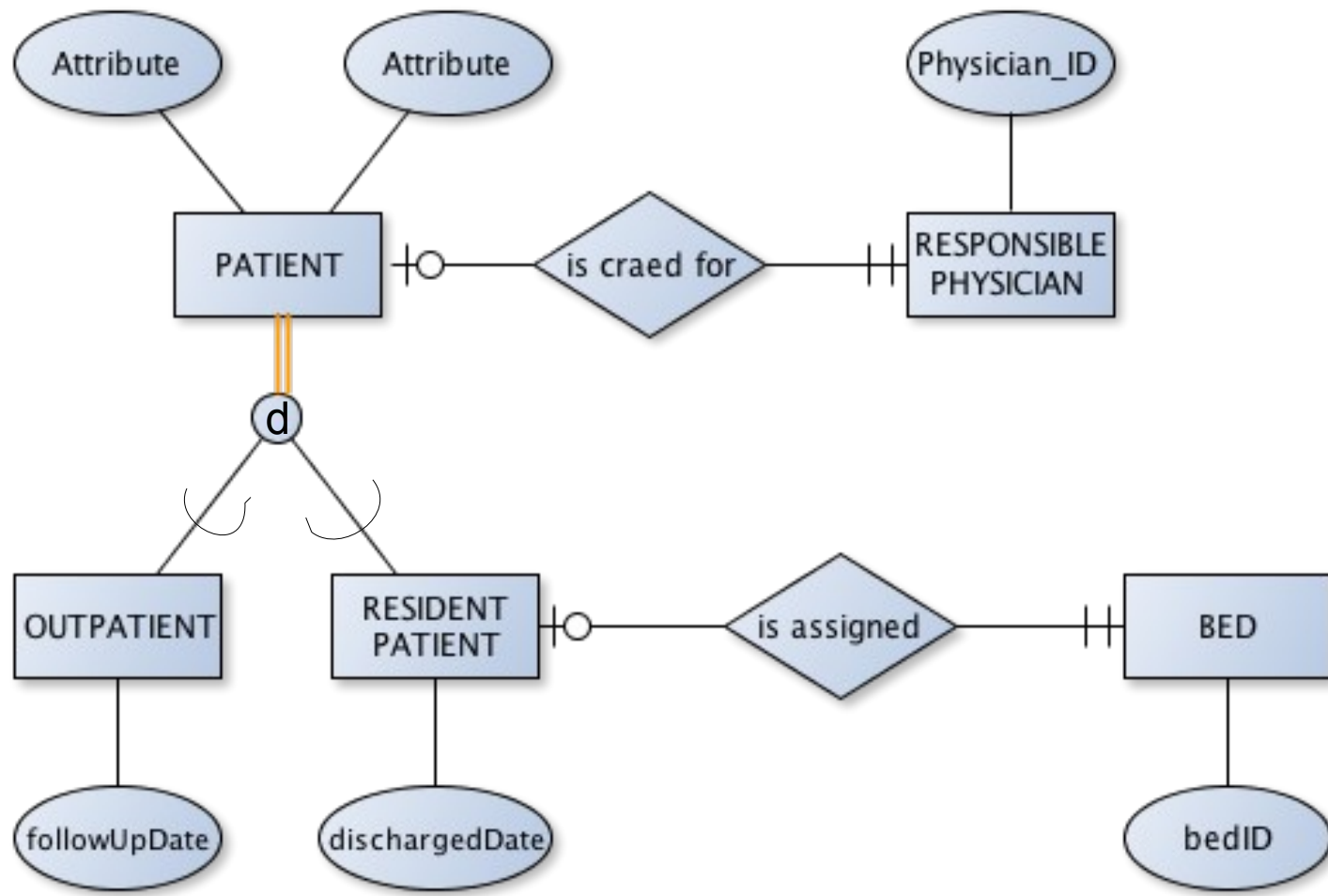
Completeness Constraints



Disjointness Constraints

- เป็นการพิจารณาว่ารายการตัวอย่างที่ปรากฏในเอนทิตีที่เป็น Supertype นั้น จะปรากฏในเอนทิตีที่เป็น Subtype ได้ 1 เอนทิตีหรือมากกว่านั้น
- แบ่งออกเป็น
 - **Disjoint Rule** เป็นกฎเกณฑ์ที่กำหนดว่ารายการตัวอย่าง (Instance) ที่ปรากฏในเอนทิตีที่เป็น Subtype หนึ่ง จะไม่ปรากฏในเอนทิตีที่เป็น subtype ใดๆ ได้อีก ใช้สัญลักษณ์คือ “d”
 - **Overlap Rule** เป็นกฎเกณฑ์ที่กำหนดว่ารายการตัวอย่าง (Instance) ที่ปรากฏในเอนทิตีที่เป็น Subtype หนึ่ง อาจจะปรากฏในเอนทิตีที่เป็น subtype อื่นด้วย ใช้สัญลักษณ์คือ “o”

Disjointness Constraints



Disjointness Constraints

