บทที่ 2

แนวเหตุผลและทฤษฎีที่สำคัญ

<u>ระบบฐานข้อมูล</u>

ฐานข้อมูล หมายถึงการรวบรวมข้อมูลที่จะใช้ร่วมกันเข้าด้วยกันโดยกำหนดรูปแบบที่แน่นอน ข้อมูลต่าง ๆ ในองค์กรนั้นจะมีการรวบรวมภายใต้การควบคุมจากศูนย์กลางเดียวกัน[1] นอกจากฐานข้อมูลประกอบไปด้วยส่วนที่เป็นข้อมูลแล้ว ยังรวมถึงปัจจัยต่าง ๆ ที่จะมีประโยชน์ต่อระบบด้วยข้อมูลที่ถูกรวบรวมเป็นฐานข้อมูลนั้นอาจจะเก็บอยู่ในรูปของการใช้คอมพิวเตอร์หรือไม่ใช้ก็ได้ แต่โดยทั่วไปเมื่อกล่าวถึงฐานข้อมูลจะหมายถึงฐานข้อมูลทางคอมพิวเตอร์[2]

1. แบบจำลองข้อมูล (Data Modeling) การกำหนดแบบจำลองข้อมูล คือการอธิบายโครงสร้างของฐานข้อมูล ได้แก่ ชนิดข้อมูล (Data Type) ความสัมพันธ์ (Relationship) และข้อกำหนดของข้อมูลนั้น แบ่งเป็น 3 ระดับ คือ[3]

1.1 ระดับบน (High Level) หรือแบบจำลองเชิงแนวคิด (Conceptual Data Model) เป็นโครงสร้างข้อมูลในระดับที่ผู้ใช้งานมองเห็น หรือเกี่ยวข้องกับผู้ใช้งานโดยตรงไม่เกี่ยวกับวิธีการจัดเก็บข้อมูล

1.2 ระดับระเบียน (Record Base) หรือแบบจำลองข้อมูลเชิงการใช้งาน (Implementation - Data Model) เป็นโครงสร้างในระดับการใช้งานสามารถเข้าใจโดยผู้ใช้ได้ แต่ไม่สัมพันธ์กับการจัดเก็บข้อมูล

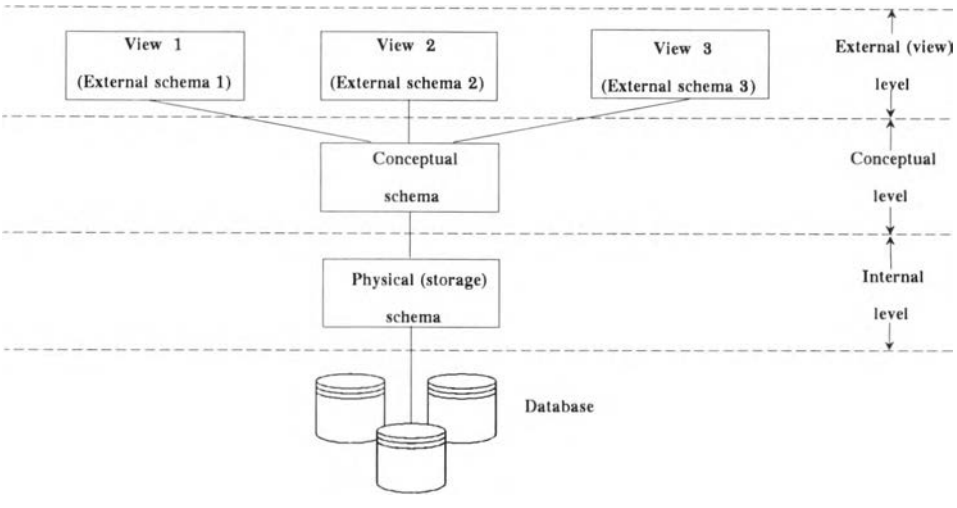
1.3 ระดับล่าง (Low Level) หรือแบบจำลองข้อมูลเชิงกายภาพ (Physical Data Model) เป็นโครงสร้างข้อมูลที่ถูกเก็บภายในสื่อต่าง ๆ ของคอมพิวเตอร์

2. สถาปัตยกรรมของข้อมูล การใช้ฐานข้อมูลจะช่วยให้ผู้ใช้งานข้อมูลได้สะดวกและไม่ต้องคำนึงถึงลักษณะการจัดเก็บ และการจัดการ เนื่องจากระบบฐานข้อมูลได้เก็บรายละเอียด วิธีการจัดเก็บ ตลอดจนการจัดการกับฐานข้อมูลไว้ นอกจากนี้ยังสามารถกำหนดขอบเขตการใช้ฐานข้อมูลตามความสำคัญของงาน และผู้ใช้งานได้ ลักษณะรูปแบบของข้อมูล (Schema) ที่สอดคล้องกับแบบจำลองที่สร้างขึ้นนั้นแบ่งเป็น 3 ระดับ คือ[4]

2.1 ระดับกายภาพ (Internal Level) เป็นระดับที่พิจารณาถึงรูปแบบการบันทึกภายในสื่อของคอมพิวเตอร์

2.2 ระดับหลักการ (Conceptual Level) หมายถึงระดับที่กำหนดรูปแบบของฐานข้อมูล ได้แก่ ชนิด ลักษณะ และความสัมพันธ์ของแบบจำลองข้อมูลเชิงตรรก นอกจากนี้ยังสามารถกำหนดขอบเขตการใช้ข้อมูลของกลุ่มผู้ใช้งานในระดับนี้ได้

2.3 ระดับนอกสุด (External Level) หมายถึงระดับของการแสดงโครงสร้างของข้อมูลในฐานข้อมูลที่ผู้ใช้งานแต่ละคนได้ถูกกำหนดไว้ให้ใช้ได้เท่านั้น ดังนั้นผู้ใช้งานแต่ละคนจะเห็นโครงสร้างข้อมูลแตกต่างกัน ลักษณะโครงสร้างข้อมูลที่แต่ละคนเห็นแตกต่างกันนี้จะเรียกว่าวิว (View)



รูปที่ 2.1 แสดงถึงสถาปัตยกรรมมาตราฐานของระบบฐานข้อมูล 3 ระดับ

3. ระบบการจัดการฐานข้อมูล (Database Management System) ระบบการจัดการฐานข้อมูล หรือเรียกย่อ ๆ ว่า DBMS นั้น ประกอบด้วยการรวมข้อมูลเข้าด้วยกันและชุดคำสั่งที่จะมาดำเนินการกับข้อมูลนั้น ซึ่งระบบการจัดการฐานข้อมูลจะอำนวยความสะดวก ประสิทธิภาพในการเรียกใช้ และการจัดเก็บข้อมูลลงในฐานข้อมูล[5]

3.1 หน้าที่ของระบบการจัดการฐานข้อมูล[6]

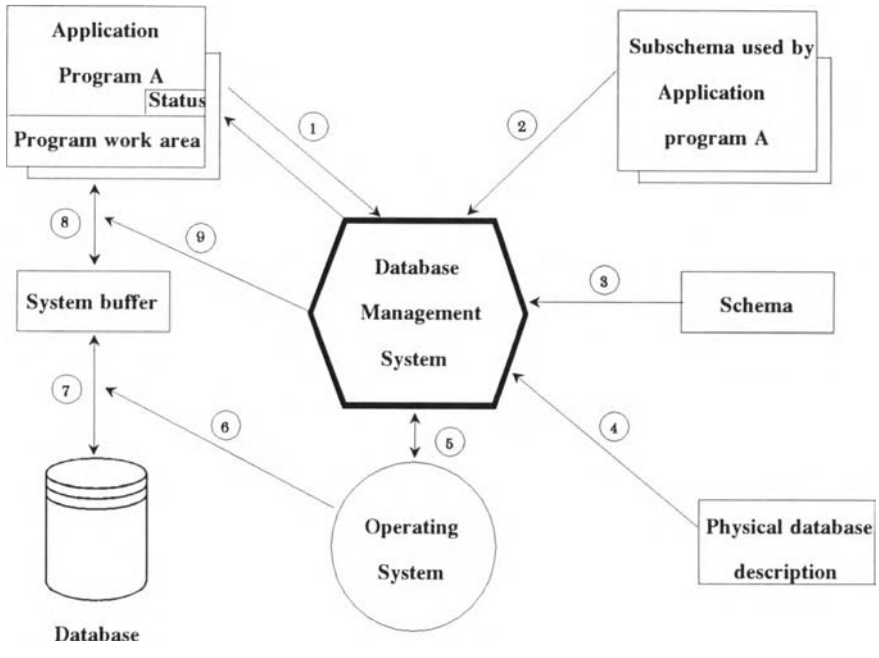
3.1.1 กำหนดลักษณะองค์ประกอบของฐานข้อมูล ได้แก่ การกำหนดองค์ประกอบของระเบียน การกำหนดชนิดของระเบียน รวมถึงการกำหนดความสัมพันธ์ต่าง ๆ

3.1.2 การกำหนดโครงสร้างการจัดเก็บของฐานข้อมูล ได้แก่ การกำหนดโครงสร้างของสื่อบันทึกข้อมูล วิธีการในการเข้าถึงข้อมูล การดึงข้อมูลมาใช้งาน การแก้ไข และการสำรองระบบข้อมูล

3.1.3 กำหนดการเข้าถึงข้อมูลภายในฐานข้อมูลด้วยภาษาทางคอมพิวเตอร์

3.1.4 จะต้องมีระบบการควบคุมได้ เช่น การสำรองข้อมูล การกู้ระบบเมื่อมีความเสียหายเกิดขึ้น รวมทั้งการควบคุมภาวะการดำเนินการที่พร้อมกันของระบบ

3.2 ขั้นตอนการทำงานของระบบการจัดการฐานข้อมูล ลักษณะของเหตุการณ์ต่าง ๆ ที่เกิดขึ้นเมื่อต้องการอ่านระเบียนจากฐานข้อมูลหนึ่ง ๆ แสดงได้ดังรูปที่ 2.2 [7]



รูปที่ 2.2 ขั้นตอนการทำงานของระบบการจัดการฐานข้อมูล

จากรูป 2.2 สามารถอธิบายการทำงานได้ดังนี้ คือ

3.2.1 เมื่อโปรแกรม A ต้องการอ่านระเบียนจากฐานข้อมูล ก็จะต้องระบุถึงรายการที่ต้องการโดยกำหนดชนิดของข้อมูล และคีย์ที่จะเข้าถึงรายการนั้น ๆ

3.2.2 ระบบการจัดการฐานข้อมูลจะค้นหาลักษณะรูปแบบของข้อมูลที่เรียกใช้โดยโปรแกรม A ในระบบของฐานข้อมูล

3.2.3 ระบบการจัดการฐานข้อมูลจะพิจารณาถึงลักษณะรูปแบบของข้อมูล โดยจะตัดสินว่าชนิดข้อมูลใดที่จำเป็น และต้องการใช้จากคำสั่งที่เรียกใช้นั้น

3.2.4 ระบบการจัดการฐานข้อมูล จะพิจารณาถึงลักษณะการจัดเก็บข้อมูลทางกายภาพและเลือกข้อมูลที่ต้องการอ่านจากสื่อที่บันทึกข้อมูล

3.2.5 ระบบการจัดการฐานข้อมูล จะสั่งให้ระบบปฏิบัติการอ่านข้อมูลจากสื่อที่บันทึกข้อมูล

3.2.6 ระบบปฏิบัติการจะค้นหาตำแหน่งที่เก็บข้อมูลบนสื่อนั้น ๆ

3.2.7 ข้อมูลที่ต้องการจะถูกเคลื่อนย้ายจากสื่อมาเก็บในที่พักของระบบ

3.2.8 ระบบฐานข้อมูลจะเลือกข้อมูลจากที่พักข้อมูลตามที่โปรแกรมต้องการ โดยลักษณะข้อมูลนี้จะเหมือนกับที่กำหนดไว้ในโครงสร้างของข้อมูล

3.2.9 ระบบการจัดการฐานข้อมูล จะเคลื่อนย้ายข้อมูลจากที่พักข้อมูลมาเก็บในพื้นที่การทำงานของชุดคำสั่งที่จัดเตรียมไว้

3.2.10 ระบบการจัดการฐานข้อมูลจะส่งข้อความที่แสดงถึงสถานะการทำงาน และข้อผิดพลาด (ถ้ามี) ไปยังโปรแกรมที่เรียกใช้งาน

3.2.11 โปรแกรมจะสามารถใช้ข้อมูลในพื้นที่การทำงานของแต่ละโปรแกรมได้

4. ชนิดของฐานข้อมูล แบ่งเป็น 3 ประเภท คือ [8]

4.1 ฐานข้อมูลแบบลำดับชั้น (Hierarchical Model) ลักษณะของโครงสร้างแบบนี้จะคล้ายกับต้นไม้ โดยที่มีการกำหนดส่วนที่เป็นแม่ และลูกไว้ตั้งแต่ตอนที่ออกแบบซึ่งที่ตัวลูกจะมีความสัมพันธ์กับแม่เพียงหนึ่งเดียวเท่านั้น

4.2 ฐานข้อมูลแบบข่ายงาน (Network Model) เป็นรูปแบบของโครงสร้างข้อมูลที่ลูกสามารถสัมพันธ์กับแม่มากกว่าหนึ่งได้ แต่ก็จะต้องมีการกำหนดความสัมพันธ์ไว้ล่วงหน้าเช่นเดียวกับแบบแรก

4.3 ฐานข้อมูลแบบสัมพันธ์ (Relational Model) เป็นโครงสร้างฐานข้อมูลที่สามารถกำหนดความสัมพันธ์ขึ้นมาเมื่อใดก็ได้ไม่จำเป็นต้องกำหนดไว้ตั้งแต่ครั้งแรก ทำให้การใช้งานมีความคล่องตัวและการเปลี่ยนแปลงทำได้ง่ายกว่ารูปแบบอื่น

5. ข้อดีในการเลือกใช้ฐานข้อมูล [9]

5.1 ระบบสามารถเปลี่ยนแปลงได้ง่าย เนื่องจากข้อมูลไม่ขึ้นกับลักษณะทางตรรก (Logical Data Independence) ซึ่งหมายถึงการที่สามารถเปลี่ยนรูปแบบข้อมูลในระดับหลักการ โดยไม่มีผลกระทบต่อระดับนอกสุด และอีกประการหนึ่งคือ ข้อมูลไม่ขึ้นต่อลักษณะทางกายภาพ (Physical Data - Independence) ซึ่งหมายถึงการที่เปลี่ยนรูปแบบทางกายภาพโดยไม่มีผลกระทบต่อรูปแบบในระดับหลักการ และระดับนอกสุด [10]

5.2 ลดความซับซ้อน ถ้าพิจารณาถึงระบบงานที่ใช้ระบบของแฟ้มข้อมูลแล้วจะเห็นว่าเป็นระบบที่อิสระต่อกัน โดยแต่ละระบบจะออกแบบมาโดยมีแฟ้มข้อมูลเป็นของตัวเอง ดังนั้นย่อมเกิดความซ้ำซ้อนของข้อมูลเกิดขึ้น ความซ้ำซ้อนอาจจะเกิดจากทีมงานที่พัฒนาระบบเป็นคนละชุด หรืออาจจะเกิดจากความเร่งด่วนในการใช้งาน ดังนั้นเมื่อใช้ระบบของฐานข้อมูลจึงนับเป็นความสำคัญอย่างยิ่งที่จะต้องให้ความสำคัญในส่วนนี้ โดยผู้ที่ออกแบบระบบงานจะต้องพิจารณาร่วมกัน และจะต้องพิจารณาร่วมกับเจ้าหน้าที่ที่บริหารข้อมูล (Database Administration - DBA) ซึ่งเจ้าหน้าที่บริหารข้อมูลนั้น จะเป็นผู้ตัดสินใจว่าจะเก็บข้อมูลภายในสื่ออย่างไร และกำหนดนโยบายต่าง ๆ ในการควบคุมการจัดเก็บ และการเรียกใช้ข้อมูล [11]

5.3 ความถูกต้องของข้อมูลมีมากขึ้น ความผิดพลาดของข้อมูลอาจจะเกิดจากความซ้ำซ้อนกันของข้อมูล การปรับปรุงแฟ้มข้อมูลจากหลายที่อาจทำให้ข้อมูลเดียวกันมีความแตกต่างกันได้ ดังนั้นถ้าควบคุมความซ้ำซ้อนได้ก็จะทำให้ปัญหาด้านความขัดแย้งของข้อมูลหมดไป

5.4 ระบบมีความปลอดภัยสูงขึ้น ระบบฐานข้อมูลสามารถควบคุมการเข้าถึง การแก้ไขเปลี่ยนแปลง การสร้างใหม่ และการกระทำการทุกอย่างบนฐานข้อมูลได้ โดยการกำหนดหลักการ และวิธีการในการเข้าถึงระบบ

5.5 ข้อมูลสามารถใช้ร่วมกันได้ ระบบงานทุกระบบที่มีความสัมพันธ์กันจะใช้ฐานข้อมูลชุดเดียวกันได้ รวมไปถึงการที่ต้องพัฒนาระบบใหม่ขึ้นมา ซึ่งทำให้พัฒนาได้เร็วยิ่งขึ้น

5.6 ระบบงานมีมาตราฐานมากขึ้น เมื่อมีฐานข้อมูลชุดเดียวกัน ระบบงานก็จะถูกควบคุมไปโดยอัตโนมัติด้วยลักษณะของการใช้ซอฟต์แวร์ในการพัฒนาระบบ และการจัดเก็บข้อมูลในสื่อ

5.7 ลดความขัดแย้งของระบบงาน เนื่องจากการออกแบบข้อมูลเป็นไปในแนวเดียวกันทั้งองค์กร ด้วยการศึกษาจากความต้องการของทุกส่วนงานในการออกแบบเบื้องต้น จึงทำให้ระบบงานขององค์กรเดียวกันมีความสัมพันธ์ไปในแนวเดียวกัน

<u>การออกแบบฐานข้อมูล</u> [12]

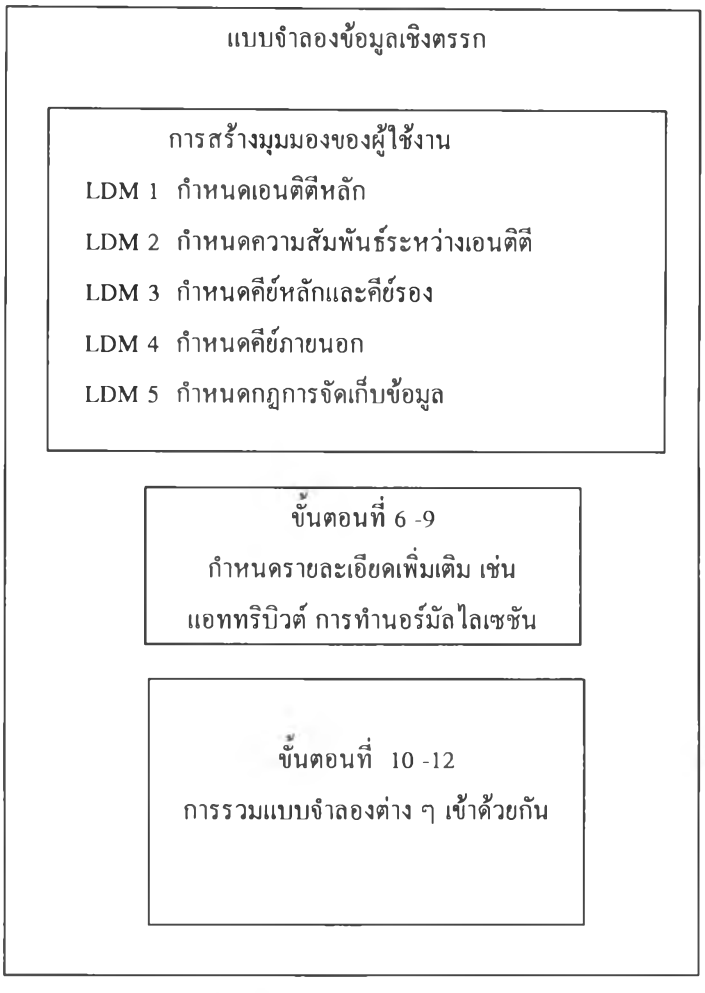
ในที่นี้จะกล่าวถึงการออกแบบฐานข้อมูลเชิงสัมพันธ์ แบ่งเป็น 2 ขั้นตอน คือการออกแบบแบบจำลองข้อมูลเชิงตรรก (Logical Data Modeling ย่อว่า LDM) และการออกแบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database Design ย่อว่า RDD)

1. แบบจำลองข้อมูลเชิงตรรก เป็นแบบจำลองที่สร้างขึ้นเพื่อแสดงโครงสร้าง และลักษณะข้อมูลที่ต้องการ เน้นลักษณะข้อมูลที่มีอยู่จริง แบบจำลองนี้จะเป็นฐานสำหรับการออกแบบ และดำเนินการในขั้นต่อไป เนื่องจากฐานข้อมูลที่ได้จากการสร้างในขั้นนี้จะแสดงถึงความสัมพันธ์ที่ถูกต้อง และสามารถนำไปใช้งานในขั้นต่อไปได้ง่ายขึ้น การออกแบบในขั้นนี้จะต้องกำหนดสิ่งที่เป็นพื้นฐาน ได้แก่ เอนติตี (Entity) ความสัมพันธ์ และแอททริบิวต์ (Attribute)

เอนติตี คือ สิ่งที่มีอยู่จริงทางกายภาพ หรือเป็นจินตภาพที่มีความหมายเป็นอย่างเดียว เมื่อกล่าวแล้วทุกคนเข้าใจเป็นที่ตรงกัน เช่น บริษัท ประเภทธุรกิจ พนักงาน นักศึกษา รายวิชา เป็นต้น

ความสัมพันธ์ หมายถึงความสัมพันธ์ระหว่างเอนติตีหนึ่งกับเอนติตีอื่น โดยอาจจะมีความสัมพันธ์กันมากกว่า 1 เอนติตีได้ เช่น นักศึกษากับรายวิชา มีความสัมพันธ์กันโดยที่นักศึกษา 1 คน สามารถลงทะเบียนมากกว่า 1 วิชาได้ หรือ บริษัทกับธุรกิจ มีความสัมพันธ์กัน คือบริษัทประกอบธุรกิจ เป็นต้น

แอททริบิวต์ คือคุณลักษณะที่เป็นรายละเอียด หรือกลุ่มความจริงที่เกี่ยวข้อง เพื่ออธิบายเอนติตี เช่น บริษัทจะประกอบด้วยชื่อบริษัท ที่ตั้ง เป็นต้น



รูปที่ 2.3 แสดงขั้นตอนการสร้างแบบจำลองข้อมูลเชิงตรรก

ขั้นตอนในการสร้างแบบจำลองข้อมูลเชิงตรรก มีดังต่อไปนี้

1.1 LDM 1 : กำหนดเอนติตีหลัก ผู้ออกแบบจะต้องศึกษาระบบให้เข้าใจเพื่อสามารถกำหนดเอนติตีที่เกี่ยวข้องกันในระบบหนึ่ง ๆ ได้

1.2 LDM 2 : กำหนดความสัมพันธ์ระหว่างเอนติตี การกำหนดความสัมพันธ์ระหว่างเอนติตี สามารถกำหนดได้โดยการบอกทิศทางความสัมพันธ์ (Direction) และการกำหนดอัตราส่วน (Cardinality Ratio)

1.2.1 ชนิดของความสัมพันธ์ มี 3 ชนิด คือ

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

1.2.1.1 ความสัมพันธ์ที่มีอยู่จริง (Existence Relationships) เช่น นิติบุคคลสามารถกำหนดธุรกิจที่จะดำเนินกิจการได้ ดังนั้นนิติบุคคลและธุรกิจ มีความสัมพันธ์ในรูปที่เป็นความสัมพันธ์ที่มีอยู่จริง คือกำหนดธุรกิจที่จะดำเนินการ

1.2.1.2 ความสัมพันธ์ตามหน้าที่ (Functional Relationships) หมายถึงความสัมพันธ์ในลักษณะหน้าที่ เช่น เอนติตีของผู้ตรวจสอบบัญชี มีความสัมพันธ์กับเอนติตีของนิติบุคคลในด้านหน้าที่ที่จะต้องตรวจสอบบัญชีของนิติบุคคลนั้น ๆ

1.2.1.3 ความสัมพันธ์ตามเหตุการณ์ที่เกิดขึ้น (Event Relationships) เช่น เมื่อนิติบุคคลเพิ่มทุนในการประกอบการก็จะต้องเกิดความสัมพันธ์ระหว่างเอนติตีของนิติบุคคลกับเอนติตีของทุนจดทะเบียน

1.2.2 อัตราส่วนของความสัมพันธ์ หมายถึงความสัมพันธ์ที่เกิดขึ้นระหว่างเอนติตีระดับพ่อแม่ (Parent Entity) และเอนติติในระดับลูก (Child Entity) โดยจะกำหนดให้ใช้ลูกศรชี้จากเอนติตีแม่ไปยังเอนติตีลูกเสมอ อัตราส่วนของความสัมพันธ์ระหว่างเอนติตี แบ่งได้ดังนี้

1.2.2.1 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (1:1 หรือ One-to-One Relationships) หมายถึง ความสัมพันธ์ของเอนติตีแม่มีความสัมพันธ์กับเอนติตีลูกเพียง 1 รายการเท่านั้น

1.2.2.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1:N หรือ One-to-Many Relationships) หมายถึง ความสัมพันธ์ที่เอนติตีแม่ สามารถสัมพันธ์กับเอนติตีลูกได้มากกว่า 1 รายการ

1.2.2.3 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (M:N หรือ Many-to-Many Relationships) หมายถึงความสัมพันธ์ที่เอนติตีแม่ สามารถมีรายการของเอนติตีลูกได้มากกว่า 1 รายการ และในทางกลับกัน เอนติตีลูกก็สามารถมีเอนติตีแม่มากกว่า 1 รายการได้ สำหรับความสัมพันธ์ในรูปแบบนี้ผู้ออกแบบจะต้องดำเนินการเปลี่ยนความสัมพันธ์นี้ ให้อยู่ในรูปความสัมพันธ์แบบ 1:N เพราะโดยทั่วไปแล้ว ฐานข้อมูลไม่สามารถจัดการกับความสัมพันธ์แบบกลุ่มต่อกลุ่มได้

1.3 LDM 3 : กำหนดคีย์หลัก และคีย์รอง เป็นขั้นตอนการเพิ่มแอททริบิวต์ที่เป็นคีย์หลักและคีย์รองของเอนติตี พร้อมทั้งกำหนดว่าคีย์ใดเป็นคีย์หลัก คีย์ใดเป็นคีย์รอง

ทุก ๆ เอนติตีจะต้องมีคีย์หลักเสมอ และมีคุณสมบัติที่ต่างจากคีย์ตัวอื่นคือ มีค่าเป็นค่าว่างไม่ได้ (Null Character) และในเอนติตีหนึ่ง ๆ อาจจะกำหนดให้มีคีย์รองได้ซึ่งคีย์รองก็มีคุณสมบัติในการอ้างอิงถึงรายการต่าง ๆ ในเอนติตีได้เหมือนคีย์หลัก แต่จะต้องเป็นคนละชุดกับคีย์หลัก สำหรับรายการใดที่แบ่งเป็น ซับไทป์ (Subtype) และซูเปอร์ไทป์ (Supertype) จะต้องระบุเอนติตีที่เป็นคีย์หลักในสองเอนติตีนั้นให้ตรงกัน

1.4 LDM 4 : กำหนดคีย์ภายนอก (Foreign Keys) คีย์ภายนอกเป็นแอททริบิวต์ของเอนติตีหนึ่ง ที่อ้างอิงถึงแอททริบิวต์ในอีกเอนติตีหนึ่ง ซึ่งลักษณะนี้ทำให้เกิดความสัมพันธ์ระหว่างสองเอนติตีขึ้น และยังมีประโยชน์ในการตรวจสอบความถูกต้องของข้อมูลด้วย ความสำคัญของคีย์ภายนอก คือ

1.4.1 ทำให้เกิดกฎธุรกิจ (Business Rules) กับเอนติตีต่าง ๆ

1.4.2 เป็นประโยชน์ในการทำให้สามารถมองเห็นได้ว่าเอนติตีใดเป็นเอนติตีลูก และเอนติตีใดเป็นเอนติตีแม่

1.4.3 ทำให้การออกแบบฐานข้อมูลง่ายขึ้น

1.5 LDM 5 : กำหนดกฎการจัดเก็บข้อมูล (Key Business Rules) เป็นขั้นตอนที่ใช้ตรวจสอบความถูกต้อง และความสอดคล้องกันของข้อมูล สามารถแบ่งกฎเป็น 3 ชนิด

1.5.1 กฎธุรกิจของคีย์ เป็นกฎที่กล่าวถึงการจัดการกับเอนติตีที่มีความสัมพันธ์ ในด้านการเพิ่ม การลบ หรือแก้ไขรายการต่าง ๆ เช่น จะยอมให้มีการเพิ่มข้อมูลในเอนติตีลูกได้ ถ้ามีข้อมูลที่สัมพันธ์กันในเอนติตีแม่ (Depentdent Rule) หรือจะยอมให้มีการลบข้อมูลในเอนติตีแม่ได้ ถ้าไม่มีรายการอ้างอิงอยู่ในเอนติตีลูก (Restricted Rule) เป็นต้น

1.5.2 โดเมน (Domain) หมายถึงการกำหนดกลุ่มของค่าที่เป็นไปได้สำหรับแอททริบิวต์หนึ่ง ๆ เช่น โดเมนของเงินที่มาลงหุ้นต้องเป็นตัวเลขเท่านั้น เป็นต้น

1.5.3 กฎการจัดการ (Triggering Operation) หมายถึงการกำหนดผลกระทบที่จะเกิดขึ้นเมื่อมีการเพิ่ม ลบ แก้ไข หรือเรียกใช้ข้อมูลต่อเอนติตีอื่น ๆ หรือแอททริบิวต์อื่นในเอนติตีเดียวกัน

1.6 LDM 6 : เพิ่มแอททริบิวต์ที่ไม่ใช่คีย์ลงในเอนติตี เมื่อกำหนดแอททริบิวต์ที่เป็นคีย์หลัก และคีย์รองแล้ว ก็สามารถเพิ่มแอททริบิวต์อื่นที่ขึ้นกับคีย์หลักลงในเอนติตีได้ ในกรณีที่แอททริบิวต์ใดสามารถสร้าง หรือคำนวณได้จากแอททริบิวต์อื่นในเอนติตีนั้น ให้กำหนดเป็นดิไรฟ์แอททริบิวต์ (Derived - Attributed) และกำหนดอักษร d ในแบบจำลองข้อมูล

1.7 LDM 7 : การตรวจสอบเอนติตีด้วยกฎนอร์มัลไลเซชัน (Normalization) หลักการนอร์มัลไลเซชันมี 5 ระดับ คือ

1.7.1 รูปแบบนอร์มัลระดับที่ 1 (First Normal Form : 1 NF) หมายถึงการไม่มีรายการใด ๆ ของเอนติตีที่ขึ้นต่อคีย์หลักมากกว่า 1 รายการ

1.7.2 รูปแบบนอร์มัลระดับที่ 2 (Second Normal Form : 2 NF) หมายถึงรูปแบบของเอนติตีไม่มีแอททริบิวต์ตัวใดที่ไม่ใช่คีย์หลักขึ้นกับบางส่วนของคีย์หลัก (Partial Dependency) และต้องอยู่ในรูปของนอร์มัลระดับที่ 1

1.7.3 รูปแบบนอร์มัลระดับที่ 3 (Third Normal Form : 3 NF) เอนติตีที่อยู่ในรูปแบบนี้ได้นั้น จะต้องเป็นแอททริบิวต์ที่อยู่ในรูปนอร์มัลระดับที่ 2 และไม่มีรายการใดที่ขึ้นกับแอททริบิวต์ที่ไม่ใช่คีย์ (Non-Key Attribute)

1.7.4 รูปแบบนอร์มัลระดับที่ 4 (Fourth Normal Form : 4 NF) เอนติตีจะต้องอยู่ในรูปของนอร์มัลระดับที่ 3 และตัวเลือกทุกตัวเป็นแคนดิเดตคีย์ (Candidate Key) เท่านั้น หรืออาจเรียกได้ว่าต้องอยู่ในรูป BCNF และมีความสัมพันธ์ไม่ขึ้นต่อกันในเชิงกลุ่ม

1.7.5 รูปแบบนอร์มัลระดับที่ 5 (Fifth Normal Form : 5 NF) เอนติตีที่อยู่ในรูปของนอร์มัลระดับที่ 4 และไม่สามารถแยกเอนติตีออกได้อีก

1.8 LDM 8 : กำหนดขอบเขตของโดเมนสำหรับแอททริบิวต์ เป็นขั้นตอนการกำหนดโดเมนของแอททริบิวต์ใด ๆ แล้วบันทึกในพจนานุกรมข้อมูล (Data Dictionary)

1.8.1 รายการที่บันทึกในพจนานุกรมข้อมูล ได้แก่

1.8.1.1 ชนิดข้อมูล (Data Type) เช่น จำนวนเต็ม ทศนิยม ตัวอักษร วันที่

1.8.1.2 ความยาว (Length) เช่น กำหนดความยาวเป็น 10 หลัก

1.8.1.3 รูปแบบข้อมูล (Format) เช่น วันที่ กำหนดให้มีรูปแบบ dd/mm/yy

1.8.1.4 ค่าของข้อมูล หรือช่วงค่าของข้อมูล (Range)

1.8.1.5 ความหมาย (Meaning) เพื่ออธิบายความหมายของแอททริบิวต์

1.8.1.6 มีค่าซ้ำกันได้หรือไม่ หรือมีความเป็นหนึ่งเดียว (Uniqueness)

1.8.1.7 มีค่าว่างได้หรือไม่ (Nullability) ยอมให้มีค่าเป็นว่างได้หรือไม่

1.8.2 กฎพิเศษสำหรับโดเมนของแอททริบิวต์บางชนิดดังนี้

1.8.2.1 แอททริบิวต์ที่เป็นคีย์หลัก ซึ่งอาจจะเป็นเพียงแอททริบิวต์เดียว หรือแอททริบิวต์ประกอบ จะต้องมีความเป็นหนึ่งเดียว และมีค่าเป็นค่าว่างไม่ได้ แต่แอททริบิวต์ย่อย ๆ แต่ละตัวที่ประกอบเป็นคีย์หลัก ไม่จำเป็นต้องเป็นหนึ่งเดียวแต่รวมแล้วต้องมีความเป็นหนึ่ง

1.8.2.2 แอททริบิวต์ที่เป็นคีย์รอง มีกฎเหมือนกับคีย์หลัก แต่มีค่าเป็นว่างได้

1.8.2.3 ดีไรฟ์แอททริบิวต์ ต้องมีค่าอยู่ในช่วงผลลัพธ์ของแอททริบิวต์ที่เป็นที่มา

1.8.2.4 คีย์ภายนอก จะต้องมีกลุ่มของโดเมนสอดคล้องกับโดเมนในเอนติตีแม่

1.8.2.5 แอททริบิวต์ที่เป็นคีย์หลักของซับไทป์เอนติตีต้องมีค่าเป็นสับเซ็ต (Subset) ของคีย์หลักของเอนติตีที่เป็นซูเปอร์ไทป์

1.9 LDM 9 : กำหนดทริกเกอร์ดำเนินการ หรือกฎธุรกิจของแอททริบิวต์ เป็นขั้นตอนการกำหนดกฎเพื่อตรวจสอบความถูกต้อง และการควบคุมผลที่จะเกิดจากการกระทำต่าง ๆ เช่น การเพิ่ม การแก้ไข การลบ รวมทั้งการค้นหารายการ โดยที่จะพิจารณาถึงผลกระทบที่จะเกิดกับเอนติตีอื่น หรือแอททริบิวต์อื่นภายในเอนติตีเดียวกัน เมื่อกำหนดเรียบร้อยแล้ว ให้บันทึกลงในพจนานุกรมข้อมูล (Data Dictionary)

1.10 LDM 10 : การรวมมุมมองของผู้ใช้เข้าด้วยกัน ในการรวมวิวหรือมุมมองของผู้ใช้เข้าด้วยกัน เพื่อให้เกิดความสมบูรณ์ของแอททริบิวต์ คือ ลดความซ้ำซ้อน และขจัดปัญหาความไม่สอดคล้องกันของข้อมูล ซึ่งอาจทำได้โดยการเพิ่มความสัมพันธ์ การรวมความสัมพันธ์ การรวมแอททริบิวต์ หรือการรวมเอนติตี และอาจรวมถึง การเพิ่มกฎทางธุรกิจด้วย

1.10.1 ข้อควรพิจารณาในการรวมเอนติตี คือ

1.10.1.1 การรวมเอนติตีที่มีคีย์หลักตัวเดียวกัน และค่าที่เป็นไปได้ของคีย์หลักเหมือนกัน จะต้องได้เอนติตีใหม่ที่มีแอททริบิวต์รวมของสองเอนติตีเดิมนั้น

1.10.1.2 ถ้าสองเอนติตีมีคีย์หลักเหมือนกันแต่ค่าที่เป็นไปได้ของคีย์หลักนั้นเป็นสับเซ็ตของกันจะรวมได้ในรูปของซูเปอร์ไทป์-ซับไทป์ โดยต้องตัดแอททริบิวต์ที่เหมือนกันในเอนติตีที่เป็นซูเปอร์ไทป์ออกจากแอททริบิวต์ที่เป็นซับไทป์

1.10.1.3 ถ้าเอนติตีสองตัวมีคีย์หลักเดียวกัน แต่มีผลไปกำหนดแอททริบิวต์ที่

ต่างกันบางตัว ให้กำหนดซูเปอร์ไทป์ขึ้นมาเพื่อให้สัมพันธ์กับเอนติตีเดิมทั้งสองตัว

1.10.1.4 การเชื่อมเอนติตีสองตัวที่คีย์หลักของตัวหนึ่งเป็นคีย์รองของอีกตัวจะได้เอนติตีใหม่ที่มีคีย์หลักตามเอนติตีตัวหนึ่ง ส่วนคีย์หลักของเอนติตีอีกตัวจะกลายเป็นคีย์รองไป และมีแอททริบิวต์รวมระหว่างสองเอนติตีเดิม และตัดแอททริบิวต์ที่ซ้ำซ้อนกันออก นอกจากนี้ยังต้องกำหนดกฎต่าง ๆ ให้เป็นไปตามกฎข้อบังคับเก่า เช่น ในกรณีที่คีย์หลักเดิมกลายมาเป็นคีย์รองในเอนติตีใหม่ก็จะต้องกำหนดให้มีค่าว่างไม่ได้

1.10.1.5 การรวมเอนติตีใด ๆ ก็ตามต้องไม่มีผลไปเปลี่ยนแปลงเอนติตีอื่นที่ไม่ได้เกี่ยวข้อง

1.10.2 สิ่งที่ควรพิจารณาในการรวมรีเลชันชิป

1.10.2.1 ให้รวมรีเลชันชิประหว่างเอนติตีที่ให้ความหมายเหมือนกันเข้าด้วยกัน โดยถ้าผลทำให้เกิดเป็นรีเลชันชิปแบบกลุ่มต่อกลุ่ม จะต้องแตกให้เป็นรีเลชันชิปแบบหนึ่งต่อกลุ่ม 2 ชุด

1.10.2.2 การรวมรีเลชันชิปใด ๆ ต้องไม่กระทบกับรีเลชันชิปอื่นที่ไม่ต้องการการเปลี่ยนแปลง นอกจากจะพิจารณาแล้วเห็นว่าควรตัดออกเนื่องจากซ้ำซ้อน หรืออาจเพิ่มขึ้นใหม่เพื่อความเหมาะสม

1.10.2.3 จากการรวมเอนติตีที่มีคีย์หลักเป็นคีย์รองของเอนติตีอีกตัว ให้ตรวจสอบคีย์ภายนอกของเอนติตีอื่น ๆ ที่อ้างมาถึงว่า ได้อ้างถึงคีย์หลักหรือคีย์รองของเอนติตีใหม่ที่ได้จากการรวมนั้น ถ้าอ้างถึงคีย์รองต้องทำการเปลี่ยนให้เป็นคีย์หลัก

1.10.2.4 เมื่อรวมมุมมองแล้ว ให้กำหนดกฎธุรกิจ สำหรับรีเลชันชิปใหม่ด้วย

1.10.3 สิ่งที่ควรพิจารณาในการรวมแอททริบิวต์

1.10.3.1 ให้ทำการรวมแอททริบิวต์ ที่มีความหมายเหมือนกันภายในเอนติตีเดียวกัน และรวมค่าที่เป็นไปได้ รวมถึงกฎการจัดการเข้าด้วยกัน และให้พิจารณาค่าที่เป็นไปได้ของแอททริบิวต์อื่นด้วยว่าเปลี่ยนไปหรือไม่

1.10.3.2 เมื่อรวมเอนติตีแล้ว ให้พิจารณาตัดแอททริบิวต์ที่เป็นดีไรฟ์ หรือตัวบ่งชี้ (Flag) ที่ไม่จำเป็นออก

1.10.3.3 หลังจากรวม ตัด หรือเพิ่มรีเลชันแล้ว ให้ทำการนอร์มัลไลซ์อีกครั้ง

1.11 LDM 11 : การรวบรวมแบบจำลองที่มีอยู่เข้าด้วยกัน เป็นขั้นตอนที่เกี่ยวข้องกับส่วนประกอบของโครงสร้างข้อมูล ในระดับแบบจำลองเชิงแนวคิดโดยรวมแบบจำลองข้อมูลเชิงตรรกกับแบบจำลองที่มีอยู่แล้ว พร้อมทั้งการพิจารณาการสร้างแบบจำลองใหม่ไปพร้อมกับการพิจารณากฎเกณฑ์ข้อบังคับเดิม ซึ่งอาจมีการใช้เอนติตี หรือรีเลชันชิปร่วมกับของเดิม หรืออาจมีการกำหนดเอนติตีใหม่ด้วย

1.12 LDM 12 : การวิเคราะห์ความมีเสถียรภาพ และความเติบโตในอนาคต ในการออกแบบแบบจำลองที่ผ่านมาจะพิจารณาถึงข้อมูลในปัจจุบัน แต่ในขั้นตอนนี้เป็นส่วนที่จะต้องพิจารณาถึงสิ่งที่อาจจะเกิดขึ้นในอนาคต เพื่อดัดแปลงแบบจำลองข้อมูลให้สามารถรองรับกับการเปลี่ยนแปลงนั้น ๆ หรือเตรียมไว้เพื่อรองรับการเปลี่ยนแปลง

2. การออกแบบฐานข้อมูลเชิงสัมพันธ์ เป็นขั้นตอนการเปลี่ยนแปลงแบบจำลองข้อมูลเชิงตรรก ให้เป็นแบบจำลองข้อมูลเชิงสัมพันธ์ เพื่อให้สอดคล้องกับระบบจัดการฐานข้อมูลทั่วไปที่ใช้บนเครื่องคอมพิวเตอร์ แบ่งเป็น ลักษณะโดยทั่วไปของฐานข้อมูลเชิงสัมพันธ์ และขั้นตอนการออกแบบฐานข้อมูล - เชิงสัมพันธ์

2.1 ลักษณะโดยทั่วไปของฐานข้อมูลเชิงสัมพันธ์ มีสิ่งที่ควรพิจารณา 2 กรณี คือ

2.1.1 โครงสร้างข้อมูล (Data Structure) เป็นส่วนที่อธิบายถึงโครงสร้างข้อมูล อันประกอบด้วย เอนติตี แอททริบิวต์ และความสัมพันธ์ของแต่ละเอนติตี การสร้างตารางความสัมพันธ์เทียบได้กับเอนติตี และการกำหนดสดมภ์ ซึ่งเทียบเท่ากับแอททริบิวต์ในแบบจำลองข้อมูลเชิงตรรก การสร้างตารางความสัมพันธ์นี้อาจจะมีการรวมตารางเข้าด้วยกัน หรือแยกตารางออกจากกัน เพื่อประโยชน์ในแง่การทำงานให้รวดเร็ว ลักษณะโครงสร้างข้อมูล คือ

2.1.1.1 สดมภ์ คือ คุณสมบัติของข้อมูลที่สามารถกำหนดขอบเขต หรือตำแหน่งได้หรืออาจกล่าวได้ว่าเป็นแอททริบิวต์ ในแบบจำลองข้อมูลเชิงตรรกนั่นเอง

2.1.1.2 แถว คือ ค่าของข้อมูลแต่ละรายการ เทียบได้กับระเบียน

2.1.1.3 ตาราง คือ กลุ่มของสดมภ์ ที่มีความหมายอธิบายถึงคุณลักษณะของตารางนั้น

2.1.2 ความเป็นบูรณภาพของข้อมูล (Data Integrity) ความเป็นบูรณภาพของข้อมูล คือการรักษาให้ข้อมูลมีความเชื่อถือ มีความถูกต้อง และป้องกันมิให้เกิดความเสียหายกับข้อมูล ความเป็นบูรณภาพของข้อมูลมีกฎเกณฑ์ความถูกต้องดังนี้

2.1.2.1 กฎความเป็นบูรณภาพของเอนติตี (Entity Integrity Rule) กำหนดให้คีย์หลักจะต้องดำรงไว้ซึ่งความเป็นหนึ่งเดียวในการอ้างอิงถึงทูเปิล (Tuple) นั่นคือในการเพิ่ม แก้ไขหรือลบข้อมูลในตาราง ค่าของคีย์หลักจะเป็นค่าว่างไม่ได้

2.1.2.2 กฎความเป็นบูรณภาพในการอ้างอิง (Referential Integrity Rule) เป็นกฎที่กล่าวถึงการอ้างอิงคือ ถ้ามีการอิงจากความสัมพันธ์หนึ่ง ไปยังความสัมพันธ์อีกอันหนึ่งจะต้องมีค่าเสมอ หรือกล่าวได้ว่า ถ้ามีความสัมพันธ์ R2 ซึ่งมีคีย์นอก (Foreign Key) ที่อ้างอิงถึงคีย์หลัก (Primary Key) ในความสัมพันธ์ R1 สำหรับทุก ๆ ค่า ของคีย์นอกในความสัมพันธ์ R 2 จะต้องมีค่าเท่ากับคีย์หลัก ในแถวใดแถวหนึ่งของความสัมพันธ์ R 1 หรือมีค่าของแอททริบิวต์ทุกตัวในคีย์นอก เป็นค่าว่าง

2.1.2.3 กฎความเป็นบูรณภาพของโดเมน (Domain Integrity Rule) หมายถึงกฎการรักษาความถูกต้องของทุก ๆ สดมภ์ในตารางความสัมพันธ์ ซึ่งกฎนี้ประกอบด้วย ชนิดของข้อมูล (Data Type) ความยาวของข้อมูล (Length) ช่วงค่าของข้อมูล (Range of Data) มีค่าซ้ำกัน หรือมีค่าว่างได้หรือไม่ การรักษาความเป็นบูรณภาพของข้อมูล จะต้องดำเนินการตลอดในทุก ๆ สภาวะของฐานข้อมูล ซึ่งเมื่อมีความไม่ถูกต้องแล้ว ระบบจัดการฐานข้อมูลจะต้องเป็นผู้ดำเนินการแก้ไข การแก้ไขทำได้หลายวิธี ขึ้นกับระบบการจัดการฐานข้อมูลแต่ละตัว เช่น ระบบจะไม่ยอมรับการเปลี่ยนแปลงแก้ไขโดยจะแสดงข้อความออกมาให้ผู้ใช้ทราบ หรือระบบจะยอมให้การแก้ไขรายการนั้น ๆ เกิดได้โดยระบบจะต้องทำการปรับให้ผลลัพธ์มีความ

คงสภาพ และถูกต้อง และประการสุดท้ายคือ ระบบจะกำหนดให้ผู้ใช้ฐานข้อมูลนั้นเป็นผู้ตัดสินใจเลือกว่าต้องการให้มีการแก้ไขข้อมูลได้หรือไม่

2.2 ขั้นตอนการออกแบบฐานข้อมูลเชิงสัมพันธ์ มีขั้นตอนที่สำคัญดังนี้

2.2.1 RDD 1 : สร้างตารางความสัมพันธ์จากเอนติตีโดยตารางแต่ละตารางจะได้จาก 1 เอนติตีเท่านั้น

2.2.2 RDD 2 : ระบุสดมภ์ในตาราง โดยกำหนดให้เอนติตีแต่ละตัวคือ 1 สดมภ์

2.2.3 RDD 3 : ปรับปรุงโครงสร้างของตารางความสัมพันธ์ให้เหมาะกับสภาพแวดล้อมของระบบจัดการฐานข้อมูล รวมถึงประสิทธิภาพในการทำงานด้วย

2.2.4 RDD 4 : ออกแบบกฎธุรกิจที่เกี่ยวข้องกับตารางความสัมพันธ์

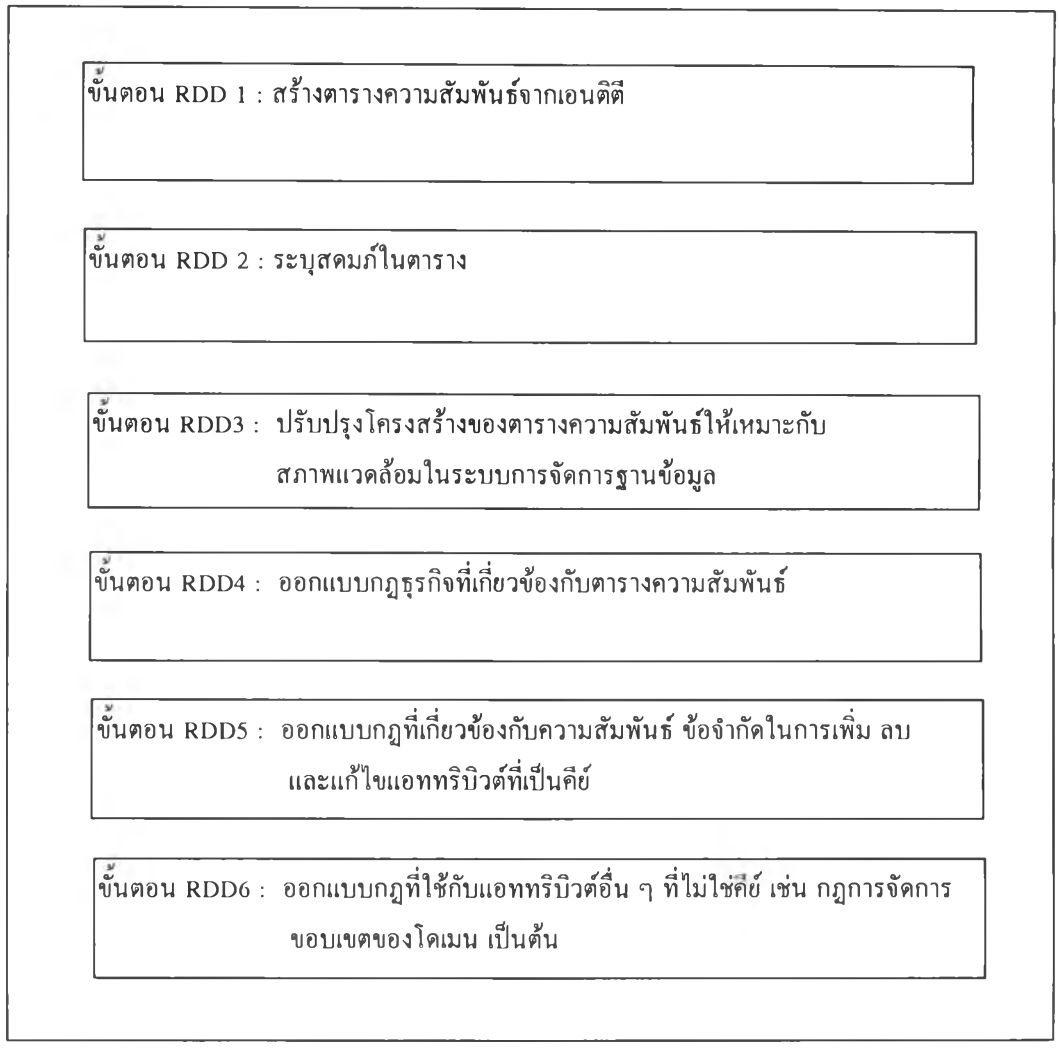
2.2.4.1 กำหนดคุณสมบัติทางตรรกของคีย์หลัก เช่น การแสดงความเป็นหนึ่งเดียว (Uniqueness) การไม่ยอมให้คีย์หลักมีค่าว่าง (Null) เป็นต้น

2.2.4.2 กำหนดคุณสมบัติทางตรรกของคีย์รอง เช่น การแสดงความเป็นหนึ่งเดียว เป็นต้น

2.2.5 RDD 5 : ออกแบบกฎที่เกี่ยวข้องความสัมพันธ์ข้อจำกัดในการเพิ่ม ลบ และแก้ไข แอททริบิวต์ที่เป็นคีย์ในลักษณะให้มีความสอดคล้องกัน

2.2.6 RDD 6 : ออกแบบกฎที่ใช้กับแอททริบิวต์อื่น ๆ ที่ไม่ใช่คีย์ได้แก่ การกำหนดกฎการจัดการ การกำหนดขอบเขตของโดเมน ชนิดข้อมูล รูปแบบข้อมูล ความเป็นเอกลักษณ์ กำหนดค่าปริยายหรือการยอมให้มีค่าว่างได้หรือไม่

ขั้นตอนการเปลี่ยนแปลงแบบจำลองข้อมูลเชิงตรรก ให้เป็นโครงสร้างฐานข้อมูลเชิงสัมพันธ์สามารถแสดงได้ดังรูป 2.4



รูปที่ 2.4 แสดงกระบวนการเปลี่ยนแบบจำลองข้อมูลเชิงตรรกเป็นโครงสร้างฐานข้อมูลเชิงสัมพันธ์