หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

แนวเหตุผลและทฤษฎีที่เกี่ยวข้อง

ระบบสารสนเทศเพื่อการจัดการ (Management Information System : MIS)

1. ความหมายของระบบสารสนเทศเพื่อการจัดการ

ความหมายกว้างๆ ของระบบสารสนเทศเพื่อการจัดการ คือ ระบบประมวลผลสารสนเทศ (Information Processing System) ซึ่งหมายถึงการนำข้อมูลมาประมวลผลเพื่อให้ได้สารสนเทศ ที่สามารถสนับสนุนการประมวลผลรายการเปลี่ยนแปลง และเป็นประโยชน์ต่อการวางแผน การตัดสินใจ และควบคุมการบริหารงานในระดับต่างๆ ขององค์กร

2. ลักษณะและโครงสร้างของระบบสารสนเทศเพื่อการจัดการ

ระบบสารสนเทศเพื่อการจัดการ จะจัดแบ่งระดับการใช้สารสนเทศเป็น 3 ระดับ คือ

2.1 สารสนเทศเพื่อการวางแผนและตัดสินใจ (Strategic Information) เป็นการใช้สารสนเทศเพื่อการวางแผน และ กำหนดนโยบายระยะยาว สำหรับผู้บริหารระดับสูง โดยสารสนเทศต้องสามารถตอบสนองความต้องการได้ทุกรูปแบบ (unstructure)

2.2 สารสนเทศเพื่อการบริหาร (Managerial Information) เป็นสารสนเทศที่ใช้ในการวางแผน และ นโยบายในการบริหาร (Tactical Planning and policy implementation)

2.3 สารสนเทศในระดับปฏิบัติการ (Operational Information) ประกอบด้วยสารสนเทศที่ใช้สนับสนุน ควบคุมการปฏิบัติการประมวลผลรายวัน ลักษณะของสารสนเทศมีการกำหนดรูปแบบแบบล่วงหน้า และระยะเวลาที่แน่นอน (Structured)

สารสนเทศเพื่อการจัดการในแต่ละระดับชั้นมีความสัมพันธ์กัน คือ ระดับล่างสุดเป็นลักษณะงานประจำการใช้สารสนเทศเพื่อการปฏิบัติงานจึงมีรูปแบบที่แน่นอน ได้จากการประมวลผลรายการเปลี่ยนแปลง ส่วนสารสนเทศในระดับการจัดการนั้น ข้อมูลที่ใช้เป็นผล

สรุปจากข้อมูลในระดับปฏิบัติการ และใช้ข้อมูลสรุปผลในระดับปฏิบัติการ รวมกับข้อมูลในระดับการจัดการและข้อมูลจากภายนอกองค์กรผลิตเป็นสารสนเทศ เพื่อให้สามารถตอบสนองความต้องการทุกรูปแบบ

3. ลักษณะและองค์ประกอบพื้นฐานของระบบสารสนเทศเพื่อการจัดการ

3.1 เป็นระบบที่เชื่อมโยงข้อมูลแบบเบ็ดเสร็จ

ความหมายของระบบเบ็ดเสร็จ (Integrated System) คือ มีการรวมข้อมูลเข้าไว้ด้วยกันเพื่อลดปัญหาความซ้ำซ้อน (Redumdance) และความไม่สอดคล้อง (Inconsistency) ของข้อมูล และให้ผู้ใช้สามารถเรียกใช้ข้อมูลร่วมกันได้

3.2 ใช้คอมพิวเตอร์ช่วยในการประมวลผล (Computer-base System)

โดยปกติระบบสารสนเทศเพื่อการจัดการ ไม่จำเป็นต้องใช้เครื่องคอมพิวเตอร์ในการประมวลผล แต่ถ้าในหน่วยงานมีการทำงานที่ซับซ้อน และ ปริมาณข้อมูลมีจำนวนมาก การดำเนินงานโดยบุคลากรอาจได้คำตอบที่ล่าช้าเกินกว่าใช้ให้เกิดประโยชน์ได้เต็มที่ ดังนั้นเพื่อให้การดำเนินงานเป็นไปอย่างมีประสิทธิภาพ จึงจำเป็นต้องนำคอมพิวเตอร์มาช่วยดำเนินงาน

3.3 มีการผสมผสานระหว่างฮาร์ดแวร์และซอฟท์แวร์ เพื่อช่วยให้ผู้ใช้มีความสะดวกในการดำเนินงาน (User-machine Interface)

3.4 จัดเตรียมสารสนเทศ ให้สามารถตอบสนองความต้องการในการบริหารงานทุกระดับ

3.5 เป็นระบบที่สนับสนุนการปฏิบัติงาน และ ช่วยในการวางแผน และตัดสินใจในการบริหารงานขององค์กร

เนื่องจากระบบสารสนเทศเพื่อการจัดการ จำเป็นต้องใช้ข้อมูลทั้งภายในและนอกหน่วยงานเพื่อนำมาประมวลผลให้ทันต่อการใช้งาน ดังนั้นจำเป็นต้องมีวิธีการจัดเก็บข้อมูลที่เหมาะสม สามารถดึงสารสนเทศมาใช้งานได้อย่างรวดเร็ว และวิธีการที่เหมาะสมในปัจจุบัน คือ การใช้ระบบฐานข้อมูล (Data Base System) เข้ามาช่วยในการจัดเก็บข้อมูล

ระบบฐานข้อมูล (Data Base System)

1. ความหมายของระบบฐานข้อมูล

ระบบฐานข้อมูล เป็นระบบที่มีจุดมุ่งหมายในการเก็บรักษาข้อมูลอย่างมีประสิทธิภาพ (Computer base Recording System) ระบบฐานข้อมูลประกอบด้วย ฐานข้อมูล (Data Base) และ ซอฟต์แวร์ (Software)

ฐานข้อมูล เป็นการเก็บบันทึกข้อมูลที่สัมพันธ์กันในลักษณะระบบเบ็ดเสร็จ กล่าวคือ มีการจัดเก็บข้อมูลไว้ส่วนกลาง เพื่อลดปัญหาความซ้ำซ้อนของข้อมูล อีกทั้งเพื่อให้ผู้ใช้ (user) สามารถเรียกใช้ข้อมูลร่วมกันได้ โดยมีโปรแกรมระบบจัดการฐานข้อมูล (Data Base Management System : DBMS) ช่วยจัดการควบคุม และอำนวยความสะดวกในการสร้าง การใช้งาน และการดูแลการเรียกใช้ข้อมูลในฐานข้อมูล

2. จุดประสงค์ของระบบฐานข้อมูล (Objective of Data Base)

2.1 ลดปัญหาความซ้ำซ้อนของข้อมูล (Reduce Data Redundance)

ระบบฐานข้อมูลมีการรวมข้อมูลไว้ที่เดียวกันและมีผู้บริหารฐานข้อมูล (Data Base Administration : DBA) รับผิดชอบดูแลข้อมูลโดยตรง ซึ่งบางครั้งอาจมีข้อจำกัดเรื่องประสิทธิภาพ อาจยอมให้มีความซ้ำซ้อนในการจัดเก็บได้ แต่สามารถควบคุมการใช้งานโดยผู้บริหารฐานข้อมูลได้

2.2 ลดปัญหาความขัดแย้งของข้อมูล

ปัญหาความไม่สอดคล้องกันของข้อมูล เกิดจากข้อมูลถูกเก็บบันทึกไว้หลายที่ ดังนั้นเมื่อมีการปรับปรุงแก้ไขข้อมูล แก้ไขไม่ครบทุกที่ ดังนั้นปัญหาดังกล่าวสามารถถูกขจัดให้ลดลงได้ เมื่อมีการลดปัญหาความซ้ำซ้อนของข้อมูล การปรับปรุงแก้ไขข้อมูลจึงกระทำเพียงแห่งเดียว

2.3 สามารถใช้ข้อมูลร่วมกันได้ (Sharing of Data)

การพัฒนาระบบข้อมูล จำเป็นต้องมีการวางแผนข้อมูลภายในองค์กรให้เป็นมาตรฐาน และเก็บบันทึกข้อมูลไว้ที่ศูนย์กลาง ดังนั้นแต่ละโปรแกรมประยุกต์สามารถเรียกใช้ข้อมูลร่วมกันได้ โดยผู้บริหารฐานข้อมูลจะเป็นผู้กำหนดขอบเขตการเรียกใช้ข้อมูลของผู้ใช้แต่ละคน

2.4 สามารถควบคุมความเป็นมาตรฐานของข้อมูล โดยที่ผู้บริหารฐานข้อมูลเป็นผู้ดูแล และกำหนดมาตรฐานข้อมูล

2.5 ข้อมูลมีความเป็นอิสระ (Data Independence)

ระบบฐานข้อมูลมีความเป็นอิสระระหว่างการจัดเก็บข้อมูล และการ

ประยุกต์ใช้งาน เพราะการใช้งานของผู้ใช้ถูกแยกเป็นอิสระจากลักษณะวิธีการจัดเก็บข้อมูล

2.6 สนองตอบความต้องการใช้ข้อมูลขององค์กร

เนื่องจากการพัฒนาฐานข้อมูล ใช้หลักการออกแบบโดยการพิจารณาข้อมูล (Data Driven) ซึ่งพิจารณาข้อมูลและความสัมพันธ์ของข้อมูลที่มีจริงในระบบ โดยดังนั้น เมื่อมีความต้องการใช้งานของโปรแกรมประยุกต์ หรือ ระบบงานใหม่ๆ จึงสามารถสนองตอบความต้องการได้ และช่วยให้การพัฒนาระบบเป็นไปได้อย่างรวดเร็ว

<u>การออกแบบระบบโดยหลักการพิจารณาข้อมูล (Data-driven System Development)</u>

การพัฒนาระบบโดยหลักการพิจารณาข้อมูล มีวัตถุประสงค์เพื่อติดตั้งระบบภายใต้แบบแผนของฐานข้อมูลแบบเบ็ดเสร็จ (Integrated Conceptual Schema) หลักการดังกล่าวนี้มีลักษณะที่สำคัญดังต่อไปนี้

1. เป็นหลักการที่อธิบายถึงความสัมพันธ์ของข้อมูล ภายใต้สภาวะแวดล้อมของฐานข้อมูล
2. เน้นที่ความเป็นอิสระระหว่างความต้องการของผู้ใช้ การติดตั้ง และการจัดการข้อมูลในระบบ
3. ระบบที่พัฒนาขึ้นมีความเป็นอิสระ ไม่ผูกติดกับระบบจัดการฐานข้อมูลใด ๆ โดยการใช้เทคนิคการทำโมเดลข้อมูล (Data Modelling)

เมื่อเปรียบเทียบการพัฒนาโดยหลักการพิจารณาข้อมูล กับ การพัฒนาระบบแบบสัญนิยม (Conventional File System) สามารถแสดงให้เห็นข้อแตกต่างกันดังนี้ คือ

	<u>การพัฒนาระบบแบบสัญนิยม</u>	<u>การพัฒนาระบบโดยหลักการพิจารณาข้อมูล</u>
1.	เป็นหลักการที่เก็บบันทึกแฟ้มข้อมูลและฐานข้อมูลแยกจากกัน	รวมฐานข้อมูลเข้าด้วยกันเป็นฐานข้อมูลแบบเบ็ดเสร็จ
2.	ในการพัฒนาระบบจะต้องรวบรวมความต้องการของผู้ใช้ที่ถูกต้อง และครบถ้วนจากผู้ใช้ก่อนจะออกแบบและสร้างระบบ	ยึดหลักการจัดทำต้นแบบ (Prototyping) เพื่อให้ผู้ใช้ค้นหาความต้องการเพิ่มเติม อีกทั้งในขั้นตอนของการพัฒนาต้นแบบ ยังได้ทำการตรวจสอบความถูกต้อง และ แก้ไขปรับปรุงโมเดลให้ถูกต้องตามความต้องการของผู้ใช้ โดยผู้ใช้ให้คำแนะนำปรึกษา

การพัฒนาระบบโดยหลักการพิจารณาข้อมูลประกอบด้วย 6 ขั้นตอนดังต่อไปนี้

1. การสำรวจ (Survey)
   - จัดทำหมายกำหนดการ และ กำหนดขอบเขตของระบบ
   - รวบรวม และทำการวิเคราะห์สารสนเทศที่เกี่ยวข้องกับข้อมูล ภายใต้ขอบเขตที่กำหนดไว้
   - จัดทำโมเดลข้อมูลเชิงตรรกระดับสูง
2. ออกแบบ (Design)
   - เป็นการพัฒนาโมเดลข้อมูลภายใต้สภาวะแวดล้อมใหม่
   - วางแผนจัดทำต้นแบบ
3. จัดทำต้นแบบ (Prototyping)
   - ออกแบบโครงสร้างฐานข้อมูลเชิงกายภาพ เพื่อจัดทำต้นแบบฐานข้อมูลในระยะแรก
   - จัดทำต้นแบบตามโครงสร้างที่ได้ออกแบบไว้
   - ใช้ต้นแบบที่จัดทำขึ้น เพื่อตรวจสอบ และแก้ไขมุมมองของผู้ใช้ให้ถูกต้อง
   - เพิ่มเติมความต้องการข้อมูลในโมเดล
   - แปลงจากต้นแบบเข้าสู่สภาวะการทำงาน (Working Status)
4. รวมระบบเข้าด้วยกัน (Integration)
   - รวมโมเดลข้อมูลเชิงตรรกที่เป็นต้นแบบเข้าด้วยกันรวมทั้งปรับจุดเชื่อมต่อระหว่างโมเดลข้อมูลเชิงตรรกให้เป็นส่วนร่วมระหว่างกัน (Inter schema mapping)
5. ปรับแต่ง (Tuning)
   - แก้ไขและปรับแต่งโครงสร้างฐานข้อมูลให้เหมาะสมเพื่อให้การทำงานมีประสิทธิภาพ
6. ตรวจทวน (Review)
   - ควบคุมคุณภาพของผลลัพธ์ที่ได้ และทำการแปลงจากต้นแบบเป็นสภาวะดำเนินงานจริง (Production Status)

สำหรับการพัฒนาระบบสารสนเทศเพื่อการบริหารแห่งชาติด้านการเงิน เป็นการออกแบบโมเดลข้อมูล โดยอาศัยหลักการที่สำคัญ 2 ประการ คือ

- โมเดลข้อมูลเชิงตรรก (Logical Data Modelling : LDM)
- การออกแบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database Design : RDD)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

โมเดลข้อมูลเชิงตรรก

1. หลักการของโมเดลข้อมูลเชิงตรรก

โมเดลข้อมูลเชิงตรรก เป็นเทคนิคใช้รูปแผนภาพ (Diagram) ร่วมกับการอธิบาย เพื่อเก็บบันทึกโครงสร้าง ความสัมพันธ์ของข้อมูลและความต้องการสารสนเทศของระบบ ซึ่งเทคนิคดังกล่าวนี้จะพิจารณาข้อมูลที่มีจริงในระบบ โดยมองข้ามการเรียกใช้งาน หรือกลุ่มผู้ใช้ และรายละเอียดในการติดตั้งระบบ

หลักการทำโมเดลข้อมูลในขั้นต้นเป็นการกำหนดโครงร่าง และ ออกแบบโมเดลข้อมูลเชิงตรรกสำหรับมุมมองของผู้ใช้ (User View) ซึ่งเป็นการพิจารณาข้อมูลที่ต้องการใช้ในการปฏิบัติงานหรือกิจกรรมแต่ละอย่างในธุรกิจ จากนั้นใช้ทฤษฎีนอร์มัลไลเซชันเข้าช่วยเพื่อจัดโครงสร้างของโมเดลให้เหมาะสม เมื่อได้โมเดลข้อมูลสำหรับแต่ละมุมมองของผู้ใช้แล้วทำการรวมโมเดลข้อมูลเชิงตรรกของมุมมองของผู้ใช้เข้าด้วยกัน เป็นโมเดลข้อมูลเชิงตรรกแบบเบ็ดเสร็จ (Integrated Logical Data Model)

2. ลักษณะที่สำคัญของโมเดลข้อมูลเชิงตรรก (Characteristic of Logical Data Model)

2.1 การทำโมเดลข้อมูลเชิงตรรกอยู่บนพื้นฐานของการพัฒนาระบบโดยหลักการพิจารณาข้อมูล

2.2 เป็นเทคนิคที่แทนความหมายของข้อมูลในระบบอย่างชัดเจน โดยการใช้รูปแผนภาพรวมกับการบรรยาย ดังนั้นจึงสามารถสื่อความหมายระหว่างผู้ออกแบบ ผู้ใช้ และผู้พัฒนาระบบได้ดี อีกทั้งยังง่ายต่อการเรียนรู้และใช้งาน

2.3 มีลักษณะที่เป็นอิสระ ไม่ผูกติดกับระบบจัดการฐานข้อมูลตัวใด ๆ ดังนั้นจึงมีความยืดหยุ่นต่อการติดตั้งระบบ

2.4 โมเดลที่ได้จากการออกแบบโดยหลักการดังกล่าว จะอยู่บนพื้นฐานของความถูกต้อง ความสอดคล้องกันของข้อมูล และสามารถใช้ข้อมูลร่วมกันได้

2.5 ยืดหยุ่นต่อการเลือกใช้เทคโนโลยีด้านฐานข้อมูล

3. โครงสร้างของโมเดลข้อมูลเชิงตรรก

โมเดลข้อมูลเชิงตรรกประกอบด้วยส่วนสำคัญคือ เอนติตี (Entity) สมาชิกในเอนติตี (Entity Instance) แอตตริบิวต์ (Attribute) และรีเลชันชิป (Relationship) โดยที่

3.1 เอนติตี (Entity)

เอนติตี คือ สิ่งต่างๆ ที่มีอยู่จริงทางกายภาพ (Physical existance) เช่น พนักงาน สินค้า หรือ เป็นจินตภาพที่มีความในตัวเอง (Conceptual existance) เช่น บริษัท รายวิชา เป็นต้น

การกำหนดเอนติตีอาศัยคุณสมบัติของสิ่งที่กำลังพิจารณาเป็นตัวกำหนด นอกจากนี้ สามารถจัดแบ่งเอนติตีออกเป็นเอนติตีแบบซับไทป์ (Subtype Entity) และ เอนติตีแบบซูเปอร์ไทป์ (Supertype Entity) โดยอาศัยคุณสมบัติบางประการของเอนติตีเป็นตัวจัดแบ่ง เช่นเอนติตีเงินกู้โดยอาศัยคุณสมบัติแหล่งที่มาของเงินกู้เป็นตัวจัดแบ่งประเภท สามารถแบ่งเงินกู้ออกเป็น 2 กลุ่มคือ เงินกู้จากแหล่งเงินกู้ภายในประเทศ และเงินกู้จากแหล่งเงินกู้ต่างประเทศ

3.2 แอตตริบิวต์ (Attribute)

แอตตริบิวต์ คือ คุณสมบัติ หรือคุณลักษณะของเอนติตี หรือ เป็นกลุ่มความจริงที่เกี่ยวข้องและอธิบายเอนติตีหรือรีเลชันชิป เช่น เอนติตีกระทรวง ประกอบด้วยแอตตริบิวต์ รหัสกระทรวง (Code]Ministry) ชื่อกระทรวง (Name]Ministry) และ วันที่เริ่มใช้งาน (Start]Date)

สามารถจัดแอตตริบิวต์ออกเป็นประเภทต่างๆได้ดังต่อไปนี้

3.2.1 แอตตริบิวต์ประเภทแคนดิเดทคีย์ (Cardidate Key Attribute)

หมายถึง แอตตริบิวต์หรือกลุ่มของแอตตริบิวต์ที่ใช้สามารถใช้อ้างอิงถึงสมาชิกแต่ละตัวในเอนติตี ซึ่งในแต่ละเอนติตี อาจมีแคนดิเดทคีย์ มากกว่า 1 ชุด ตัวอย่าง เช่น เอนติตี กระทรวง ทั้ง รหัสกระทรวง และ ชื่อกระทรวง ต่างก็เป็นแคนดิเดทคีย์

3.2.2 แอตตริบิวต์ประเภทคีย์หลัก (Primary Key Attribute)

หมายถึง แอตตริบิวต์หรือกลุ่มของแอตตริบิวต์ที่สามารถใช้เป็นตัวอ้างอิงสมาชิกแต่ละตัวในเอนติตี หรือกล่าวได้ว่าเป็นแคนดิเดทคีย์ ที่ถูกเลือกให้เป็นคีย์หลักของเอนติตี ซึ่งแอตตริบิวต์ที่เป็นคีย์หลัก จะต้องมีค่าไม่เป็นค่าว่าง (Null)

3.2.3 แอตตริบิวต์ประเภทคีย์รอง (Alternate Key Attribute)

หมายถึง แอตตริบิวต์หรือกลุ่มของแอตตริบิวต์ที่สามารถใช้เป็นตัวอ้างอิงถึงสมาชิกแต่ละตัวในเอนติตี เพียงแต่ไม่ได้รับการคัดเลือกให้เป็นคีย์หลักของเอนติตี หรือกล่าวได้ว่า เป็นแคนดิเดทคีย์ ที่ไม่ได้ถูกคัดเลือกให้เป็นคีย์หลักของเอนติตี

3.2.4 แอตตริบิวต์ประเภทฟอร์เรนจ์คีย์ (Foriegn Key Attribute) หมายถึง แอตตริบิวต์หรือกลุ่มของแอตตริบิวต์ที่เป็นตัวเชื่อมความสัมพันธ์ระหว่างเอนติตี โดยที่แอตตริบิวต์ดังกล่าวจะเป็นคีย์หลักของเอนติตีหนึ่ง และไปปรากฏในอีกเอนติตี เช่น เอนติตี กระทรวง ประกอบด้วย แอตตริบิวต์ รหัสกระทรวง ชื่อกระทรวง วันที่เริ่มใช้งาน คีย์หลัก คือ รหัสกระทรวง และ เอนติตีกรม ประกอบด้วย แอตตริบิวต์ รหัสกระทรวง รหัสกรม ชื่อกรม วันที่เริ่มใช้งาน คีย์หลักของเอนติตีกรม คือ รหัสกระทรวง+รหัสกรม ฟอร์เรนจ์คีย์เชื่อมความสัมพันธ์ระหว่างเอนติตีกระทรวงและกรม คือ รหัสกระทรวง คือ แอตตริบิวต์รหัสกระทรวง

3.2.5 แอตตริบิวต์ประเภทไม่ใช่คีย์ (Non Key Attribute) หมายถึง แอตตริบิวต์ที่ไม่ได้เป็นส่วนประกอบของคีย์หลัก

3.2.6 แอตตริบิวต์ประเภทดิไรฟว์ (Derived Attribute) หมายถึง แอตตริบิวต์ที่มีค่าขึ้นกับค่าแอตตริบิวต์อื่น

3.3 รีเลชันชิป (Relationship)

รีเลชันชิป คือ ความสัมพันธ์ที่เกี่ยวข้องกันระหว่าง 2 เอนติตีขึ้นไป โดยรีเลชันชิปประกอบด้วย ทิศทางของความสัมพันธ์ (Direction) และสัดส่วนความสัมพันธ์ (Cardinality Ratio)

ทิศทางความสัมพันธ์จะเป็นตัวบ่งชี้ทิศทางความสัมพันธ์จากเอนติตีแม่ไปยังเอนติตีลูกและสัดส่วนความสัมพันธ์เป็นการประมาณอัตราส่วนจำนวนสมาชิกที่สัมพันธ์กันระหว่าง 2 เอนติตี

โดยอาศัยสัดส่วนความสัมพันธ์ สามารถจัดแบ่งรีเลชันชิปได้ 3 ลักษณะดังนี้

3.3.1 รีเลชันชิปแบบหนึ่งต่อหนึ่ง (1:1 or One-to-One Relationship) เป็นความสัมพันธ์ระหว่างสมาชิกในเอนติตีแม่ 1 ตัว สัมพันธ์กับสมาชิกในเอนติตีลูก 1 ตัว เช่นรีเลชันชิประหว่างค่าผลิตภัณฑ์มวลรวมประชาชาติภายในประเทศ (Gross Domestic Product : GDP) กับปีงบประมาณ โดยแต่ละปีจะมีค่าค่าผลิตภัณฑ์มวลรวมประชาชาติภายในประเทศ 1 ค่า

3.3.2 รีเลชันชิปแบบหนึ่งต่อกลุ่ม (1:N or One-to-Many Relationship) เป็นความสัมพันธ์ระหว่างสมาชิกในเอนติตีแม่ 1 ตัว สัมพันธ์กับสมาชิกในเอนติตีลูกหลายตัว เช่น รีเลชันชิประหว่างเอนติตีกระทรวง และ เอนติตีกรม โดยที่แต่ละกระทรวงประกอบด้วยหลายกรม

3.3.3 รีเลชันชิปแบบกลุ่มต่อกลุ่ม (N:M or Many-to-Many Relationship) เป็นความสัมพันธ์ระหว่างสมาชิกในเอนติตีแม่ 1 ตัว สัมพันธ์กับสมาชิก

ในเอนติตีลูกหลายตัว และจำนวนสมาชิกในเอนติตีลูก 1 ตัว สัมพันธ์กับสมาชิกในเอนติตีแม่หลายตัว เช่น รีเลชันชิประหว่างแผนงานกับกระทรวง โดยที่กระทรวงหนึ่งมีแผนงานที่รับผิดชอบหลายแผนงาน และแผนงานหนึ่งรับผิดชอบโดยหลายหน่วยงาน

4. แผนภาพโมเดลข้อมูลเชิงตรรก (Logical Data Model Diagram)

การจัดทำรูปแผนภาพโมเดลข้อมูลเชิงตรรก จะแทนเอนติตีด้วยสี่เหลี่ยม และมุมบนซ้ายของสี่เหลี่ยมเป็นชื่อของเอนติตี ภายในสี่เหลี่ยมประกอบด้วยชื่อแอตตริบิวต์ และความสัมพันธ์ระหว่างเอนติตี แสดงโดยใช้เส้นตรงลากเชื่อมต่อระหว่างสี่เหลี่ยม และทิศทางของความสัมพันธ์แทนด้วยหัวลูกศร ซึ่งหัวลูกศรเดี่ยวหมายถึงจำนวนสมาชิกเป็น 1 และหัวลูกศรคู่หมายถึงจำนวนสมาชิกเป็นกลุ่ม

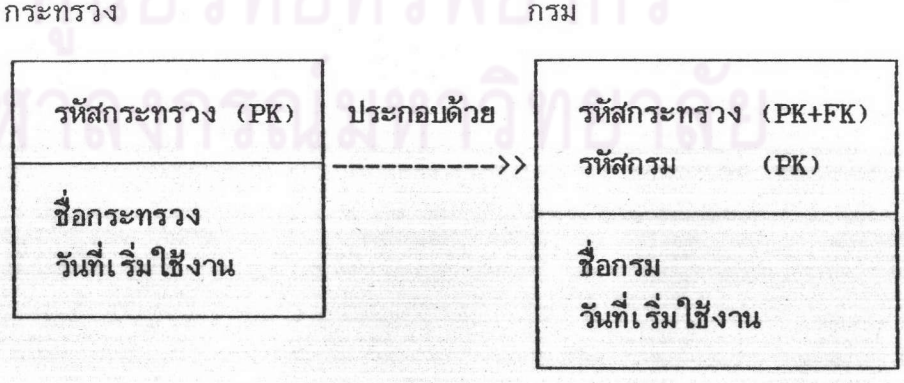
---------->	แทนรีเลชันชิปแบบหนึ่งต่อหนึ่ง
---------->>	แทนรีเลชันชิปแบบหนึ่งต่อกลุ่ม
<<-------->>	แทนรีเลชันชิปแบบกลุ่มต่อกลุ่ม

ภายในสี่เหลี่ยมจะถูกแบ่งด้วยเส้นตรง โดยเหนือเส้นตรงเป็นรายชื่อของแอตตริบิวต์ที่ถูกเลือกให้เป็นคีย์หลักของเอนติตี และใต้เส้นแบ่งเป็นรายชื่อของแอตตริบิวต์ที่ไม่ใช่คีย์ หลังชื่อแอตตริบิวต์อาจมีสัญญลักษณ์กำกับ ซึ่งมีความหมายดังนี้

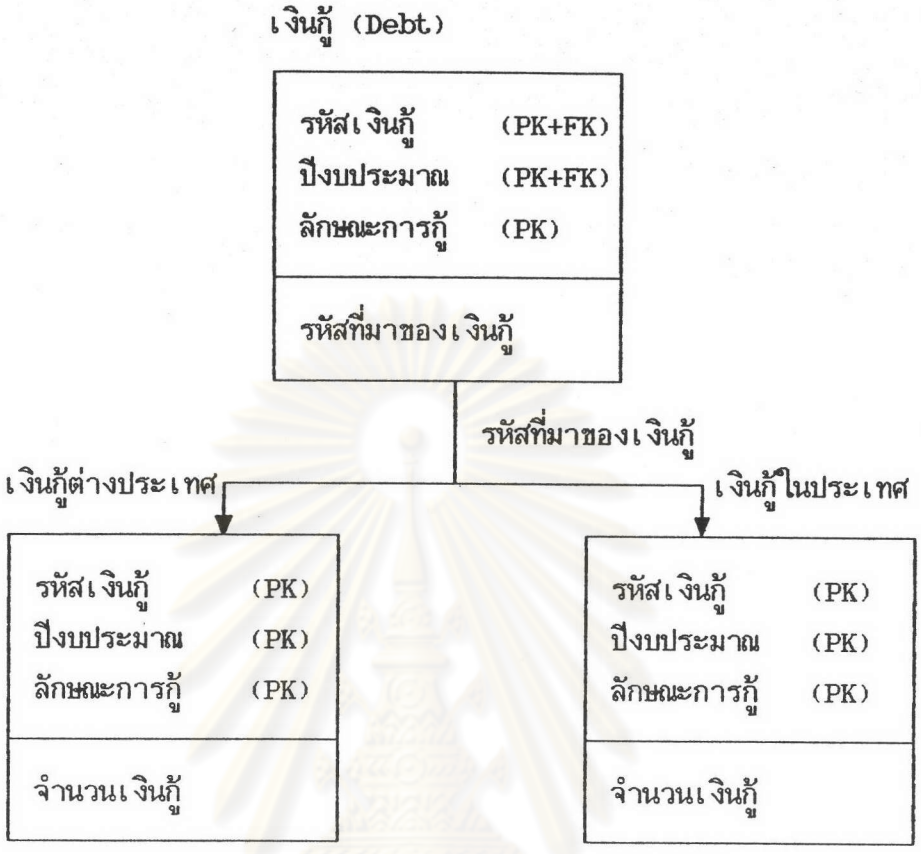
(PK) : หมายถึงแอตตริบิวต์ดังกล่าวเป็นส่วนประกอบของคีย์หลัก

(FK) : หมายถึงแอตตริบิวต์ดังกล่าวเป็นส่วนประกอบของฟอร์เรนจ์คีย์

ตัวอย่างของแผนภาพดังรูป



รูปที่ 2.1 แสดงแผนภาพรีเลชันชิปแบบหนึ่งต่อกลุ่มระหว่างเอนติตีกรมและกระทรวง



รูปที่ 2.2 แสดงแผนภาพรีเลชันชิประหว่างเอนติตีแบบซับไทป์และซูเปอร์ไทป์

5. ขั้นตอนการทำโมเดลข้อมูลเชิงตรรก (Stage in Logical Data Model)

หลักการทำโมเดลข้อมูลเชิงตรรกโดยคร่าวๆ สามารถแบ่งออกได้ เป็น 3 ส่วนหลักๆ ดังต่อไปนี้

5.1 การสร้างโครงร่างสำหรับแต่ละมุมมองของผู้ใช้

5.2 เพิ่มรายละเอียดโมเดลข้อมูลเชิงตรรกสำหรับมุมมองของผู้ใช้

5.3 รวมมุมมองของผู้ใช้เข้าด้วยกันเป็นโมเดลข้อมูลเชิงตรรกแบบเบ็ดเสร็จ

ในการจัดทำโมเดลข้อมูลเชิงตรรกดำเนินการใต้ภายหลักเกณฑ์ซึ่งมีด้วยกันทั้งหมด 12 ข้อ [Fleming,1989] โดยในแต่ละขั้นตอนแทนด้วย LDM_n ซึ่งหมายถึงการพัฒนา โมเดลข้อมูลเชิงตรรกขั้นตอนที่เอ็น ในแต่ละขั้นตอนประกอบด้วยรายละเอียดโดยสังเขปดังต่อไปนี้

LDM 1 : กำหนดเอนติตีหลัก (Identify Major Entity)

วิธีการ คือ พิจารณาจากข้อมูลในระบบที่ผู้ใช้สนใจ ซึ่งทำได้โดยการ

- รวบรวมจากเอกสารภายในหน่วยงาน
- สังเกตและสอบถามจากการปฏิบัติงาน
- ออกแบบสอบถามผู้เกี่ยวข้องภายในหน่วยงาน
- รวบรวมความคิดเห็นจากผู้พัฒนาระบบ

LDM 2 : กำหนดรีเลชันชิประหว่างเอนติตี

พิจารณารูปแบบรีเลชันชิประหว่างเอนติตี และ ทำการปรับรีเลชันชิปให้อยู่ในรูปของรีเลชันชิประหว่าง 2 เอนติตี แบบหนึ่งต่อหนึ่ง หรือหนึ่งต่อกลุ่ม ทำการบันทึกชื่อ นิยาม ของเอนติตีและรีเลชันชิป ไว้ในพจนานุกรม พร้อมทั้งจัดทำรูปแผนภาพโมเดลข้อมูลเชิงตรรก

LDM 3 : กำหนด คีย์หลัก และ คีย์รอง เพื่อระบุคุณสมบัติของเอนติตี

LDM 4 : กำหนด ฟอร์เรนจ์คีย์ เพื่อระบุคุณสมบัติของรีเลชันชิป

LDM 5 : กำหนดกฎคีย์ทางธุรกิจ (Key Business Rule) สำหรับแอตตริบิวต์

กฎเกณฑ์ทางธุรกิจ (Business Rules)

กฎเกณฑ์ทางธุรกิจ เป็นการกำหนดเงื่อนไขสำหรับการปฏิบัติการที่เกี่ยวข้องกับแอตตริบิวต์ สามารถจำแนกกฎเกณฑ์ทางธุรกิจออกได้ 3 ประเภทดังนี้

1. กฎคีย์ทางธุรกิจ (Key Business Rule)

เป็นตัวกำหนดความคงสภาพของรีเลชันชิป ซึ่งข้อกำหนดดังกล่าวนี้มีผลต่อการดำเนินการเรื่อง การเพิ่ม การลบ และแก้ไข คีย์หลัก หรือฟอร์เรนจ์คีย์ กล่าวได้ว่าเป็นข้อกำหนดความคงสภาพของข้อมูลในโมเดลข้อมูลเชิงตรรก เพื่อให้แน่ใจในความถูกต้องและสอดคล้องกันของข้อมูล

กฎคีย์ทางธุรกิจที่จะกล่าวถึงคือกฎการเพิ่มข้อมูล (Insert Rule) และกฎการลบข้อมูล (Delete Rule)

1.1 กฎการเพิ่มข้อมูล

เป็นการกำหนดเงื่อนไขที่ต้องตรวจสอบสำหรับเอนติตีแม่ ในการเพิ่ม หรือ แก้ไขฟอร์เรนจ์คีย์ของข้อมูลในเอนติตีลูก โดยทั่วไปมี 6 ลักษณะ คือ

1. การเพิ่มแบบขึ้นต่อกัน (Dependent) อนุญาติให้เพิ่มข้อมูลในเอนติตีลูกได้ถ้ามีข้อมูลที่สัมพันธ์กันในเอนติตีแม่

2. การเพิ่มแบบอัตโนมัติ (Automatic) การเพิ่มข้อมูล ถ้า

ตรวจสอบพบว่าไม่มีข้อมูลที่สัมพันธ์กันในเอนติตีแม่ ให้เพิ่มค่าข้อมูลในเอนติตีแม่

3. การเปลี่ยนค่าเป็นค่าว่าง (Nullity) การเพิ่มข้อมูล ถ้าตรวจสอบพบว่าไม่มีข้อมูลที่สัมพันธ์กันในเอนติตีแม่ ให้เปลี่ยนค่า ฟอร์เรนจ์คีย์ เป็นค่าว่าง

4. เปลี่ยนเป็นค่าที่กำหนด (Default) การเพิ่มข้อมูล กรณีที่ไม่มีข้อมูลในเอนติตีแม่ ให้เปลี่ยนค่า ฟอร์เรนจ์คีย์ เป็นค่าที่กำหนดไว้

5. แบบมีเงื่อนไข (Customize) เพิ่มข้อมูลได้ต่อเมื่อเงื่อนไขตรงตามที่กำหนด

6. ไม่มีผลกระทบ (No effect) การเพิ่มข้อมูลไม่ต้องตรวจสอบ

1.2 กฎการลบข้อมูล

เป็นการกำหนดเงื่อนไขในการแก้ไข หรือลบคีย์หลักในเอนติตีแม่ที่ถูกอ้างอิงถึงโดยฟอร์เรนจ์คีย์ในเอนติตีลูก โดยทั่วไปมี 6 ลักษณะ คือ

1. การลบแบบมีข้อจำกัด (Restric) การลบข้อมูลทำได้ เมื่อไม่มีข้อมูลที่อ้างอิงถึง

2. การลบแบบกระทำเป็นทอด ๆ (Cascade) การลบข้อมูลให้ทำการลบข้อมูลในเอนติตีลูกที่อ้างอิงถึงด้วย

3. การเปลี่ยนค่าเป็นค่าว่าง (Nullity) การลบข้อมูล จะทำการเปลี่ยนข้อมูลของฟอร์เรนจ์คีย์ที่อ้างอิงถึง ให้เป็นค่าว่าง

4. เปลี่ยนเป็นค่าที่กำหนด (Default) การลบข้อมูลจะทำการเปลี่ยนค่าข้อมูลของฟอร์เรนจ์คีย์ที่อ้างอิงถึง ให้เป็นค่าที่กำหนด

5. แบบมีเงื่อนไข (Customize) การลบข้อมูลจะกระทำได้ต่อเมื่อเป็นไปตามเงื่อนไขที่กำหนดไว้

6. ไม่มีผลกระทบ (No effect) การลบข้อมูลไม่ต้องทำการตรวจสอบเงื่อนไขใดๆ ทั้งสิ้น

2. กฎโดเมนทางธุรกิจ (Domain Business Rule)

โดเมน (Domain) หมายถึง กรอบค่าต่างๆที่เป็นไปได้ของแอตตริบิวต์ เช่นโดเมนของแอตตริบิวต์ วันที่ จะกินความเฉพาะวันที่อยู่ระหว่าง 1 ถึง 31 และ เดือน มีค่าระหว่าง 1 ถึง 12 โดเมนของชื่อ ต้องเป็นตัวอักษรเท่านั้น

3. ทริกเกอร์ดำเนินการ (Trigger Operation)

เป็นข้อกำหนดถึงผลกระทบที่ต้องดำเนินการกับเอนติตีอื่นหรือแอตตริบิวต์อื่นในเอนติตีเดียวกัน เมื่อมีการดำเนินการเกี่ยวกับ การเพิ่ม ลบ แก้ไข และเรียกใช้ข้อมูล

LDM 6 : เพิ่มแอตตริบิวต์ที่ไม่ใช่คีย์

LDM 7 : ตรวจสอบความถูกต้องของมุมมองผู้ใช้โดยใช้ทฤษฎีนอร์มัลไลเซชัน

ทฤษฎีนอร์มัลไลเซชัน (Normalization)

นอร์มัลไลเซชันเป็นทฤษฎีที่ใช้วิเคราะห์และจัดโครงสร้างข้อมูลใหม่เพื่อลดความซ้ำซ้อนของโครงสร้างข้อมูล และได้โครงสร้างที่มีเสถียรภาพ โดยการออกแบบระบบให้อยู่ในรูปนอร์มัล (Normal Form) และก่อนที่กล่าวถึงรูปนอร์มัล ขอกล่าวถึงนิยามที่เกี่ยวข้องดังต่อไปนี้

1. ตัวเลือก (Determinant) คือ แอตตริบิวต์หรือกลุ่มของแอตตริบิวต์ใด ๆ ที่สามารถเลือก (Determine) แอตตริบิวต์ตัวอื่นๆได้

2. ฟังก์ชันการขึ้นต่อกัน (Functional Dependency)

นิยาม ถ้าให้ B และ A เป็นแอตตริบิวต์ กล่าวได้ว่า B มีฟังก์ชันการขึ้นอยู่กับค่าของ A ก็ต่อเมื่อ สามารถใช้ A เป็นตัวเลือก (Determain) ค่าของ B ได้เพียงหนึ่งค่าเสมอ

สัญญลักษณ์ที่ใช้แทน B มีฟังก์ชันการขึ้นอยู่กับ A คือ A -> B

3. การขึ้นต่อกันอย่างสมบูรณ์ (Full Functional Dependency)

นิยาม ให้ B และ A เป็นกลุ่มแอตตริบิวต์ กล่าวได้ว่า B มีฟังก์ชันการขึ้นอยู่กับค่าของ A อย่างสมบูรณ์ ก็ต่อเมื่อ B มีฟังก์ชันการขึ้นอยู่กับค่าของ A แต่ไม่มีคุณสมบัติฟังก์ชันการขึ้นอยู่กับแอตตริบิวต์อื่นที่เป็นส่วนประกอบของ A

4. การขึ้นต่อกันเชิงกลุ่ม

นิยาม ในเอนติตีที่ประกอบด้วยแอตตริบิวต์ A B และ C กล่าวได้ว่าการขึ้นต่อกันเชิงกลุ่มระหว่าง B และ A โดยที่ B ขึ้นต่อ A คือค่าของ A หนึ่งค่าจะอิงกับกลุ่มของ B โดยการขึ้นต่อกันนี้จะเป็นอิสระกับค่าของ C

หลักการนอร์มัลไลเซชันโดยสังเขป คือ จัดโครงสร้างข้อมูลให้อยู่ในรูปนอร์มัลซึ่งประกอบด้วยระดับต่างๆ ดังต่อไปนี้

1. รูปนอร์มัลระดับที่ 1 (First Normal Form : 1 NF)

นิยาม เอนติตีจะอยู่ในรูปนอร์มัลระดับที่ 1 ถ้าไม่มีกลุ่มของแอตตริบิวต์ที่มีค่าซ้ำกัน (Repeating or Multivalued Attribute)

2. รูปนอร์มัลระดับที่ 2 (Second Normal Form : 2 NF)

นิยาม เอนติตีจะอยู่ในรูปนอร์มัลระดับที่ 2 เมื่อเอนติตีดังกล่าวอยู่ในรูปนอร์มัลระดับที่ 1 และ แอตตริบิวต์ที่ไม่ใช่คีย์ ต้องมีค่าขึ้นกับค่าคีย์หลักของเอนติตีอย่างสมบูรณ์

3. รูปนอร์มัลระดับที่ 3 (Third Normal Form : 3 NF)

นิยาม เอนติตีจะอยู่ในรูปนอร์มัลระดับที่ 3 เมื่อเอนติตีดังกล่าวอยู่ในรูปนอร์มัลระดับที่ 2 และทุกค่าของแอตตริบิวต์ที่ไม่ใช่คีย์ต้องไม่ขึ้นกับค่าของแอตตริบิวต์ที่ไม่ใช่คีย์อื่นๆ

4. รูปนอร์มัลแบบบอยซ์/คอดด์ (Boyce / Codd Normal Form)

นิยาม เอนติตีอยู่ในรูปนอร์มัลแบบบอยซ์/คอดด์ ก็ต่อเมื่อเอนติตีดังกล่าวอยู่ในรูปนอร์มัลระดับที่ 3 และตัวเลือก (Determinant) ทุกตัวต้องเป็นแคนดิเดตคีย์เท่านั้น

5. รูปนอร์มัลระดับที่ 4 (Fourth Normal Form : 4 NF)

นิยาม เอนติตีอยู่ในรูปนอร์มัลแบบบอยซ์/คอดด์ ก็ต่อเมื่อเอนติตีอยู่ในรูปนอร์มัลแบบบอยส์คอดด์ และเป็นความสัมพันธ์ที่ไม่ขึ้นต่อกันเชิงกลุ่ม

6. รูปนอร์มัลระดับที่ 5 (Fifth Normal Form : 5 NF)

นิยาม เอนติตีอยู่ในรูปนอร์มัลระดับที่ 5 ก็ต่อเมื่อเอนติตีดังกล่าวอยู่ในรูปนอร์มัลระดับที่ 4 และ ไม่สามารถแยกเอนติตีดังกล่าวออกได้อีก

สำหรับการออกแบบโมเดลข้อมูลเชิงตรรกโดยการประยุกต์ใช้ทฤษฎีนี้ ถึงแม้ว่าจะได้โครงสร้างที่ถูกขจัดความซ้ำซ้อน และมีเสถียรภาพ แต่เมื่อแปลงอยู่ในรูปฐานข้อมูลเชิงกายภาพอาจมีปัญหาในเรื่องประสิทธิภาพในการประมวลผล จึงอาจต้องมีการปรับให้อยู่ในรูปแบบที่เหมาะสมในภายหลัง นอกจากนี้การที่จะให้แน่ใจว่าระบบที่ได้ถูกต้องตามความต้องการของผู้ใช้ การใช้ทฤษฎีนอร์มัลไลเซชันควรกระทำภายหลังจากการออกแบบผ่านขั้นตอน LDM 1 ถึง LDM 6 มาก่อน

ปกติการจัดโครงสร้างข้อมูลให้อยู่ในรูปนอร์มัลระดับที่ 3 ก็เพียงพอ ในการขจัดปัญหาความซ้ำซ้อนและไม่สอดคล้องต้องกันของข้อมูล ส่วนการจัดให้อยู่ในรูปนอร์มัลระดับสูงขึ้นต้องพิจารณาถึงความจำเป็นของข้อมูล

LDM 8 : กำหนดโดเมนสำหรับแอตตริบิวต์ พร้อมทั้งเก็บบันทึกรายละเอียดในพจนานุกรมข้อมูล ประกอบด้วยรายละเอียด

- ชนิดของข้อมูล (Data Type)
- รูปแบบของแอตตริบิวต์ (Format)
- ช่วงค่าที่เป็นไปได้ (Range)
- ความหมาย (Meaning)
- มีค่าซ้ำกันได้หรือไม่ (Uniqueness)
- มีค่าเป็นค่าว่างได้หรือไม่ (Null Support)

หลักการในการกำหนดโดเมน

คีย์หลัก โดเมน คือต้องมีค่าไม่ซ้ำกัน และ ต้องไม่เป็นค่าว่าง
คีย์รอง โดเมน ต้องมีค่าไม่ซ้ำกัน และ ค่าเป็นค่าว่างได้
ฟอร์เรนจ์คีย์ โดเมนต้องสอดคล้องกับคีย์หลักในเอนติตีแม่
โดเมนของเอนติตีแบบซับไทป์เหมือนกับโดเมนของซูเปอร์ไทป์

LDM 9 : กำหนดทริกเกอร์ดำเนินการ

LDM 10 : รวมมุมมองของผู้ใช้เข้าด้วยกัน

การรวมมุมมองของผู้ใช้เข้าด้วยกัน มีจุดประสงค์เพื่อลดความซ้ำซ้อน ความไม่สอดคล้องของข้อมูล และเพิ่มรีเลชันชิประหว่างมุมมองของผู้ใช้ ซึ่งประกอบด้วย

1. การรวมเอนติตี
2. การรวมรีเลชันชิป
3. การรวมแอตตริบิวต์

LDM 11 : รวมโมเดลที่ได้เข้ากับโมเดลที่มีอยู่เดิม

LDM 12 : บันทึกความเปลี่ยนแปลงในอนาคตที่อาจมีผลกระทบต่อโมเดลข้อมูล

การออกแบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database Design)

การออกแบบฐานข้อมูลเชิงสัมพันธ์ เป็นการแปลจากโมเดลข้อมูลเชิงตรรกให้อยู่ในรูปของฐานข้อมูลเชิงสัมพันธ์ (Relation Data Base)

ฐานข้อมูลเชิงสัมพันธ์ประกอบด้วย โครงสร้างข้อมูล (Data Structure) และความคงสภาพของข้อมูล (Data Integrity)

1. โครงสร้างของโมเดลข้อมูลเชิงสัมพันธ์ ประกอบด้วย ตารางความสัมพันธ์ (Relational Table) คอลัมน์ (Column) และ แถว (Row) โดยที่

1.1 คอลัมน์ คือ คุณสมบัติของข้อมูล เมื่อเปรียบเทียบกับโมเดลข้อมูลเชิงตรรก คือ แอตตริบิวต์ในเอนติตี นั้นเอง

1.2 แถว คือ ค่าข้อมูลในตาราง

1.3 ตารางความสัมพันธ์ เป็นตาราง 2 มิติที่มีคุณสมบัติ 6 ประการดังต่อไปนี้

(1) ต้องไม่มีคอลัมน์ที่ซ้ำกันในตาราง
(2) ค่าข้อมูลในคอลัมน์จะต้องมีคุณสมบัติเช่นเดียวกัน
(3) ข้อมูลใน ตารางจะต้องไม่มีข้อมูลแถวใดๆ ที่ซ้ำกัน
(4) ลำดับของคอลัมน์ไม่มีความสำคัญ
(5) ลำดับข้อมูลแต่ละแถวไม่มีความสำคัญ
(6) ชื่อคอลัมน์ในตารางต้องไม่ซ้ำกัน

2. ความคงสภาพข้อมูล (Data Integrity)

เมื่อกล่าวถึงความคงสภาพของข้อมูลโดยทั่วไป จะมีความหมายครอบคุม 3 ลักษณะ คือ ความคงสภาพของเอนติตี (Entity Integrity Rule) ความคงสภาพของการอ้างอิง (Referential Integrity Rule) ความคงสภาพของโดเมน (Domain Integrity Rule)

2.1 ความคงสภาพของเอนติตี เป็นกฎสำหรับการเพิ่ม ปรับปรุง และลบข้อมูลในตาราง นิยาม คือ แอตตริบิวต์ทุกตัวที่เป็นส่วนของคีย์หลัก ต้องไม่มีค่าเป็นค่าว่าง

2.2 ความคงสภาพของการอ้างอิง

นิยาม คือ ถ้าตารางของความสัมพันธ์มีฟอร์เรนจ์คีย์ กล่าวได้ว่าทุกค่าของฟอร์เรนจ์คีย์ต้องไม่เป็นค่าว่าง หรือ มีค่าตรงกับค่าในตารางที่ฟอร์เรนจ์คีย์เป็นคีย์หลักในตารางนั้น ตัวอย่างเช่น ความสัมพันธ์ R2 มีฟอร์เรนจ์คีย์ FK ที่ใช้ในการอ้างอิงถึงคีย์หลัก PK ในความสัมพันธ์ R1 สำหรับทุกค่าของ FK ใน R2 จะต้อง

(1) มีค่าเท่ากับ PK ในความสัมพันธ์ R1
(2) มีค่าของแอตตริบิวต์ทุกตัวใน FK เป็นค่าว่าง

ความสำคัญของกฎการการคงสภาพในอ้างอิง คือเมื่อมีการอ้างอิงจากความสัมพันธ์ที่ 1 ไปยังความสัมพันธ์ที่ 2 ต้องแน่ใจว่าข้อมูลในความสัมพันธ์ที่ 2 ต้องมีค่าเสมอ

2.3 ความคงสภาพของโดเมน หมายถึง กฎการคงสภาพสำหรับคอลัมน์ทุกคอลัมน์ในตารางรวมทั้ง คีย์หลัก ฟอร์เรนจ์คีย์ และคอลัมน์ที่ไม่ใช่คีย์ ซึ่งกฎของโดเมนประกอบด้วย

(1) ชนิดของข้อมูล (Data Type)
(2) ความยาว (Length)
(3) ช่วงค่าข้อมูล (Range Checking)
(4) ค่าที่กำหนดไว้ (Default Value)
(5) มีค่าซ้ำกันได้หรือไม่ (Uniquness)
(6) มีค่าเป็นค่าว่างได้หรือไม่ (Nullability)

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

3. ลักษณะที่สำคัญของการออกแบบฐานข้อมูลเชิงสัมพันธ์

ลักษณะที่สำคัญของการออกแบบฐานข้อมูลเชิงสัมพันธ์ ประกอบด้วย

(1) ใช้หลักการออกแบบโดยการพิจารณาข้อมูล (Data-driven)

เนื่องจากโมเดลได้จากการแปลงโมเดลข้อมูลเชิงตรรก และรวมกฎเกณฑ์ทางธุรกิจเข้าในขั้นตอนการออกแบบฐานข้อมูล

(2) ใช้หลักการพิจารณากระบวนการ (Process Oriented) เพื่อให้การออกแบบสนองตอบในเรื่องประสิทธิภาพ เช่น เวลาในการตอบสนอง

(3) โมเดลมีลักษณะเป็นอิสระจากผลิตภัณฑ์ (Product Independent) ทำให้สามารถเลือกใช้ระบบจัดการฐานข้อมูลแบบสัมพันธ์ตัวใดก็ได้

4. ขั้นตอนการออกแบบโมเดลข้อมูลเชิงสัมพันธ์

ในการออกแบบโมเดลข้อมูลเชิงสัมพันธ์ ประกอบด้วย 6 ขั้นตอน โดยในแต่ละขั้นตอนแทนด้วย RDD_N หมายถึงขั้นตอนที่เอ็น

RDD 1 : ทำการแปลงเอนติตีของโมเดลข้อมูลเชิงตรรก ให้อยู่ในรูปตาราง โดย 1 เอนติตี ต่อ 1 ตาราง

RDD 2 : แปล 1 แอตตริบิวต์ของโมเดลข้อมูลเชิงตรรกเป็น 1 คอลัมน์

RDD 3 : ปรับโครงสร้างของโมเดลที่ได้จากการแปล RDD 1 และ RDD 2 ให้เหมาะกับการติดตั้งระบบตามผลิตภัณฑ์ที่เลือกใช้

RDD 4 : ออกแบบกฎเกณฑ์ธุรกิจสำหรับเอนติตี

RDD 5 : กำหนดกฎเกณฑ์ธุรกิจสำหรับความสัมพันธ์

RDD 6 : กำหนดกฎเกณฑ์ธุรกิจสำหรับแอตตริบิวต์ที่เหลือ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย