

บทที่ 10 การออกแบบฐานข้อมูล

วัตถุประสงค์ของบทเรียน

- ศึกษาเกี่ยวกับการออกแบบฐานข้อมูลที่ประสบความสำเร็จ
- ศึกษาเกี่ยวกับกระบวนการออกแบบระบบเทคโนโลยีสารสนเทศที่มีฐานข้อมูลเป็นส่วนหนึ่งของระบบ (Software Development Life Cycle, SDLC)
- ศึกษาเกี่ยวกับกระบวนการออกแบบฐานข้อมูล (Database Life Cycle, DBLC)
- ศึกษาเกี่ยวกับวิธีการประเมินและปรับปรุง SDLC และ DBLC
- ศึกษาเกี่ยวกับกลยุทธ์ในการออกแบบฐานข้อมูลในรูปแบบต่างๆ เช่น top-down, bottom-up, centralized และ decentralized

เนื้อหาของบทเรียน

ระบบเทคโนโลยีสารสนเทศ กระบวนการพัฒนาระบบเทคโนโลยีสารสนเทศ ขั้นตอนการออกแบบฐานข้อมูล การออกแบบกรอบความคิด การเลือกซอฟต์แวร์ DBMS การออกแบบเชิงตรรกะ การออกแบบเชิงกายภาพ กลยุทธ์ในการออกแบบฐานข้อมูล การออกแบบในลักษณะ centralized และ decentralized

กิจกรรมการเรียนรู้-การสอน

- อธิบายพร้อมยกตัวอย่างประกอบ
- ศึกษาจากเอกสารคำสอน
- ฝึกปฏิบัติการตามที่มอบหมาย
- ทำแบบฝึกหัดท้ายบท

อุปกรณ์ที่ใช้ในการเรียนรู้-การสอน

- เอกสารคำสอน
- เครื่องคอมพิวเตอร์
- เครื่องฉายภาพสไลด์

การวัดและประเมินผล

- การตอบคำถามระหว่างการเรียนรู้-การสอน
- การทำแบบทดสอบย่อยท้ายบท
- การตรวจงานตามที่มอบหมาย

ณ ปัจจุบันบริษัท องค์กร และ/หรือห้างร้านต่างๆได้มีการประยุกต์ใช้ระบบเทคโนโลยีสารสนเทศ (information system) สำหรับการดำเนินการทางธุรกิจต่างๆซึ่งโดยส่วนใหญ่ของระบบเทคโนโลยีสารสนเทศจะมีการประยุกต์ใช้ฐานข้อมูลเพื่อจัดเก็บข้อมูลต่างๆ ด้วยเหตุนี้ เราจึงควรที่จะต้องให้ความสนใจหรือใส่ใจกับการออกแบบฐานข้อมูล เนื่องจากถ้าเราสามารถออกแบบฐานข้อมูลได้ดีจะส่งผลให้ระบบเทคโนโลยีสารสนเทศสามารถจัดการกับข้อมูลสารสนเทศได้อย่างถูกต้อง ดังนั้น ในบทนี้เราจะทำการศึกษาเกี่ยวกับขั้นตอนการออกแบบฐานข้อมูลที่เรียกว่า “Database Life Cycle, DBLC” และ ขั้นตอนการออกแบบระบบเทคโนโลยีสารสนเทศที่เรียกว่า “Systems Development Life Cycle, SDLC” ที่จะเป็นกระบวนการทำซ้ำของกระบวนการสร้าง ตรวจสอบ ปรับปรุงพัฒนา และการปรับเปลี่ยนระบบเทคโนโลยีสารสนเทศให้สามารถทำงานมีประสิทธิภาพ

10.1 ระบบเทคโนโลยีสารสนเทศ

โดยปกติของการประยุกต์ใช้ฐานข้อมูลมักถูกใช้เป็นส่วนหนึ่งของระบบเทคโนโลยีสารสนเทศที่ซึ่งจะให้บริการในด้านการจัดเก็บและการค้นคืนข้อมูล แต่สำหรับระบบเทคโนโลยีสารสนเทศจะมีวัตถุประสงค์ที่จะอำนวยความสะดวกให้แก่ผู้ใช้งานในการปรับเปลี่ยนข้อมูลให้กลายเป็นสารสนเทศ และจะมีฟังก์ชันต่างๆ สำหรับจัดการ/จัดเก็บข้อมูลและสารสนเทศ ระบบเทคโนโลยีสารสนเทศจะประกอบไปด้วยบุคคลในหน้าที่ต่างๆ ฮาร์ดแวร์ ซอฟต์แวร์ ฐานข้อมูล แอปพลิเคชัน/โปรแกรม และกระบวนการต่างๆ

ในการที่จะทำการตัดสินใจที่จะสร้างระบบเทคโนโลยีสารสนเทศ เราควรที่จะต้องทำการสำรวจความต้องการและความจำเป็นในการใช้งานระบบเทคโนโลยีสารสนเทศ (เรียกกระบวนการนี้ว่า “system analysis”) เมื่อเราเห็นว่าบริษัทหรือองค์กรต้องการที่จะใช้งานระบบเทคโนโลยีสารสนเทศ เราจึงเริ่มกระบวนการสร้างระบบเทคโนโลยีสารสนเทศ (เรียกกระบวนการนี้ว่า “system development”)

ภายใต้กรอบการทำงานระบบเทคโนโลยีสารสนเทศจะเป็นการปรับเปลี่ยนข้อมูลให้กลายเป็นสารสนเทศที่พร้อมสำหรับสนับสนุนการตัดสินใจ โดยสารสนเทศที่ได้จะอยู่ในรูปแบบของรายงาน ตาราง รูปภาพ ชาร์ตและกราฟต่างๆ ดังแสดงในรูป 10.1 ที่จะแสดงถึงแอปพลิเคชันที่จะประกอบไปด้วย 2 ส่วนหลักๆคือ ข้อมูล และโค้ดหรือคำสั่งของโปรแกรมที่ใช้ในการปรับเปลี่ยนข้อมูลให้กลายเป็นสารสนเทศเพื่อนำไปใช้ในกระบวนการและการดำเนินการต่างๆ จากความสำคัญของระบบเทคโนโลยีสารสนเทศเราควรจะทำ การสร้างระบบเทคโนโลยีสารสนเทศที่มีประสิทธิภาพที่ซึ่งจะขึ้นกับ 3 ปัจจัยดังนี้

- การออกแบบและการดำเนินการสร้างฐานข้อมูล
- การออกแบบและการดำเนินการสร้างแอปพลิเคชัน
- กระบวนการดูแลรักษาระบบเทคโนโลยีสารสนเทศ

จากปัจจัยทั้ง 3 ข้างต้น เนื้อหาในวิชานี้จะมุ่งเน้นที่การออกแบบและการดำเนินการสร้างฐานข้อมูลที่จะเป็นปัจจัยที่สำคัญที่สุดใน 3 ปัจจัย แต่อย่างไรก็ตามถ้าเราไม่สนใจหรือใส่ใจกับ 2 ปัจจัยที่เหลืออาจทำให้

ระบบเทคโนโลยีสารสนเทศมีฟังก์ชันการทำงานที่ไม่ดีและอาจไม่สอดคล้องกับการดำเนินการทางธุรกิจก็เป็นได้ การพัฒนาระบบเทคโนโลยีจะเป็นงานที่ยากที่จะประกอบไปด้วยกระบวนการสำรวจความต้องการและความจำเป็นในการใช้งานระบบเทคโนโลยีสารสนเทศ และกระบวนการสร้างระบบเทคโนโลยีสารสนเทศ ซึ่งจะต้องการวางแผนที่ดีที่จะทำให้แน่ใจว่าทุกกิจกรรม/การดำเนินการจะสอดคล้องกับการดำเนินการทางธุรกิจและทำให้แน่ใจได้ว่าการสร้างระบบเทคโนโลยีสารสนเทศจะสำเร็จตรงเวลา

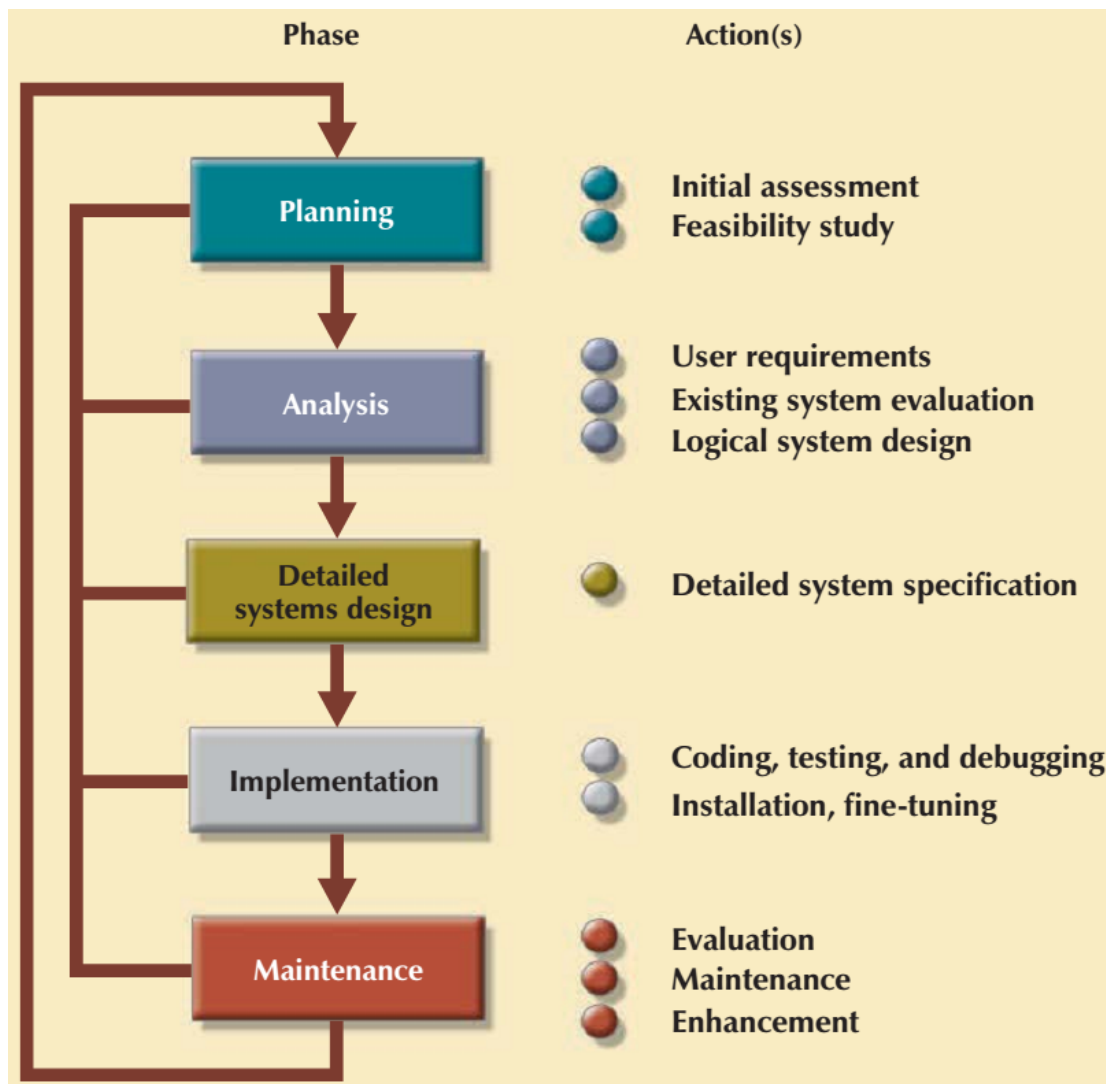


รูปที่ 10.1 การสร้างข้อมูลสารสนเทศเพื่อสนับสนุนการตัดสินใจ

10.2 กระบวนการพัฒนาระบบเทคโนโลยีสารสนเทศ

กระบวนการพัฒนาระบบเทคโนโลยีสารสนเทศ (System Development Life Cycle, SDLC) จะเป็นกระบวนการในการสร้างและติดตามการทำงานของระบบเทคโนโลยีสารสนเทศที่จะทำให้ผู้พัฒนาระบบได้เห็นภาพรวมของการออกแบบฐานข้อมูลและการสร้างแอปพลิเคชันที่ซึ่งจะทำให้เราสามารถประเมินการทำงานของระบบเทคโนโลยีสารสนเทศได้

ดังแสดงในรูป 10.2 ขั้นตอนการออกแบบระบบเทคโนโลยีสารสนเทศจะแบ่งออกเป็น 5 กระบวนการย่อย: ขั้นตอนการวางแผน (planning), ขั้นตอนการวิเคราะห์ (analysis), ขั้นตอนการออกแบบและพิจารณารายละเอียด (detailed systems design), ขั้นตอนการสร้างระบบ (implementation), และขั้นตอนการดูแลรักษาระบบ (maintainance) ตามลำดับ ขั้นตอน SDLC จะเป็นการทำงานแบบทำซ้ำ ตัวอย่างเช่น ในการทำงานรอบแรกเราอาจทำการศึกษาขั้นตอนการทำงานของบริษัท/องค์กรจากนั้นทำการลงรายละเอียดการออกแบบเบื้องต้นจากนั้นทำการประเมินว่ามีส่วนใดที่ยังไม่สมบูรณ์อีกหรือไม่แล้วทำการพิจารณาใหม่หลายๆรอบจนกระทั่งได้ระบบเทคโนโลยีสารสนเทศที่มีการดำเนินงานครบถ้วนที่สามารถตอบสนองต่อการดำเนินธุรกิจต่างๆได้



รูปที่ 10.2 กระบวนการพัฒนาระบบเทคโนโลยีสารสนเทศ

10.2.1 ขั้นตอนการวางแผน

ขั้นตอนการวางแผนจะทำให้ผู้พัฒนาระบบเห็นภาพรวมของวัตถุประสงค์และการดำเนินงานขององค์กรที่จะทำให้เราทราบถึงความต้องการที่จะได้รับสารสนเทศ จากนั้นเราจะต้องทำการประเมินความต้องการดังกล่าวที่จะช่วยให้เราสามารถตอบคำถามที่สำคัญเหล่านี้ได้

- เราควรที่จะใช้ระบบที่มีอยู่ต่อไปหรือไม่?—ถ้าระบบเทคโนโลยีสารสนเทศเดิมสามารถสร้างสารสนเทศที่เพียงพอต่อการดำเนินธุรกิจอยู่แล้วจะทำให้ไม่มีความจำเป็นที่จะต้องปรับเปลี่ยนหรือการหาระบบใหม่มาแทนที่การทำงานของระบบเดิมเลย
- เราควรที่จะทำการปรับเปลี่ยนระบบที่มีอยู่หรือไม่?—จากการประเมินจะชี้ให้เห็นถึงข้อบกพร่องของระบบที่ใช้งานอยู่ในปัจจุบันที่จำเป็นจะต้องแก้ไข ดังนั้นเราควรที่จะทำการปรับเปลี่ยนตามความจำเป็นเหล่านั้น

- เราควรที่จะหาระบบใหม่มาแทนที่ระบบที่มีอยู่หรือไม่?—จากการประเมินจะชี้ให้เห็นว่าระบบที่ใช้งานอยู่ปัจจุบันมีข้อบกพร่องและข้อบกพร่องเหล่านั้นสามารถแก้ได้ยาก ด้วยเหตุนี้จึงมีความต้องการที่จะทำการสร้างระบบการดำเนินงานใหม่ แต่อย่างไรก็ตาม เราควรที่จะสามารถแยกแยะระหว่างความต้องการและความจำเป็นในการสร้างระบบใหม่ด้วยเช่นกัน

ในการที่จะตอบคำถามข้างต้น ผู้ที่มีส่วนร่วมกับกระบวนการ SDLC จะต้องทำการศึกษาและประเมินทางเลือกอื่นๆว่าสามารถแก้ไขปัญหที่เกิดขึ้นกับระบบการทำงานเดิมได้หรือไม่ ถ้าผลประเมินบ่งบอกว่าบริษัท/องค์กรมีความต้องการและมีความจำเป็นในการพัฒนาระบบการดำเนินงานใหม่จะก่อให้เกิดคำถามที่ตามมาคือมีความเป็นไปได้หรือไม่ในการที่จะพัฒนาระบบใหม่ เพื่อที่จะตอบคำถามดังกล่าวเราควรที่จะต้องพิจารณาถึงปัจจัยต่างๆดังนี้

- ปัจจัยทางด้านฮาร์ดแวร์และซอฟต์แวร์—เราต้องพิจารณาฮาร์ดแวร์ที่จำเป็นต้องใช้หลังจากการสร้างระบบเทคโนโลยีสารสนเทศ อาทิเช่น desktop computer, multiprocessor computer, mainframe หรือ supercomputer เป็นต้น แต่ในส่วนของซอฟต์แวร์เราจะต้องคำนึงถึงการประยุกต์ใช้ single หรือ multiuser operating systems, ชนิดของฐานข้อมูลและซอฟต์แวร์ฐานข้อมูลที่ใช้, ภาษาที่ใช้ในการเขียนโปรแกรมสำหรับพัฒนาระบบเทคโนโลยีสารสนเทศ และอื่นๆ
- ปัจจัยทางด้านค่าใช้จ่าย—ในการที่จะลงทุนแต่ละครั้งของบริษัทไม่ว่าจะเป็นลงทุนในด้านเครื่องจักรหรือระบบคอมพิวเตอร์ เราควรที่จะต้องทำการศึกษาว่าจำนวนเงินที่ลงทุนนั้นนั้นมีปริมาณมากน้อยเพียงใดและจำนวนเงินที่ลงทุนมีความคุ้มค่าต่อสิ่งที่ได้รับมากน้อยเพียงใด
- ปัจจัยทางการดำเนินงาน—ในการที่จะใช้งานระบบเทคโนโลยีสารสนเทศที่สร้างขึ้น บริษัท/องค์กรต้องใช้พนักงาน เทคนิค และปริมาณเงินมากน้อยเพียงใด นอกจากนั้นเรายังต้องทำการประเมินเกี่ยวกับการจัดการต่างๆของระบบที่เกี่ยวข้องกับการดูแลรักษาระบบให้คงอยู่

10.2.2 ขั้นตอนการวิเคราะห์

ปัญหาที่พบในกระบวนการวางแผนจะถูกพิจารณาโดยละเอียดในขั้นตอนการวิเคราะห์ซึ่งจะต้องพิจารณาถึงปัจจัยดังต่อไปนี้

- อะไรคือสิ่งที่ผู้ใช้งานระบบต้องการเพิ่มเติมจากระบบปัจจุบัน?
- ความต้องการเหล่านั้นเหมาะสมและมีความสำคัญต่อความต้องการสารสนเทศในการดำเนินงานหรือไม่?

จากคำถามข้างต้น ขั้นตอนการวิเคราะห์จะต้องตรวจสอบความต้องการของผู้ใช้งานระบบโดยละเอียด การตรวจสอบจะเป็นการทำงานร่วมกันระหว่างผู้ใช้งานระบบและผู้ออกแบบระบบที่จะทำการระบุถึงขั้นตอนหรือกระบวนการต่างๆที่ก่อให้เกิดปัญหา จากการทำงานร่วมกันดังกล่าวจะทำให้สามารถระบุถึงสิ่งที่ต้องการและระบุถึงความจำเป็นในการพัฒนาระบบใหม่ได้

นอกเหนือจากการศึกษาและตรวจสอบเกี่ยวกับความต้องการเพิ่มเติมของผู้ใช้งานระบบแล้ว ขั้นตอนการวิเคราะห์จะต้องทำการออกแบบระบบเชิงตรรกะ (logical system design) ที่จะเป็นการออกแบบแบบจำลองกรอบความคิด (conceptual data model) อินพุต ขั้นตอน/กระบวนการต่างๆ และผลลัพธ์ที่คาดว่าจะได้รับการดำเนินการต่างๆ โดยในขณะทำการออกแบบระบบเชิงตรรกะ ผู้ออกแบบควรอาจประยุกต์ใช้เครื่องมือต่างๆ เช่น data flow diagram (DFD), hierarchical input process output (HIPO) diagram, แลพ entity relationship (ER) diagram เพื่อแสดงให้เห็นถึงขั้นตอนกระบวนการและข้อมูลที่เกี่ยวข้อง

10.2.3 ขั้นตอนการออกแบบและพิจารณารายละเอียด

ขั้นตอนการออกแบบและพิจารณารายละเอียดจะเป็นขั้นตอนที่จะทำให้เราสามารถออกแบบเกี่ยวกับกระบวนการทำงานต่างๆ ของระบบได้แล้วเสร็จ การออกแบบจะเป็นการกำหนดถึงหน้าจอ เมนู รายงาน และส่วนต่างๆ ที่จะทำให้ผู้ใช้งานระบบสามารถได้รับสารสนเทศได้อย่างถูกต้องและมีประสิทธิภาพ กระบวนการนี้จะเป็นการปรับเปลี่ยนกระบวนการต่างๆ จากระบบเก่าไปสู่ระบบใหม่ด้วยเช่นกัน

10.2.4 ขั้นตอนการสร้างระบบ

ในระหว่างขั้นตอนการสร้างระบบ ฮาร์ดแวร์ ซอฟต์แวร์ DBMS และแอปพลิเคชันต่างๆ จะถูกติดตั้ง นอกจากนั้นยังรวมถึงการสร้างฐานข้อมูลในซอฟต์แวร์ DBMS และอย่างที่เรารู้กันว่ากระบวนการสร้างระบบจะเป็นการบวนการเขียนโปรแกรม ตรวจสอบและแก้ไขข้อผิดพลาดต่างๆ จนกระทั่งระบบที่พัฒนาขึ้นมีความสมบูรณ์ หลังจากการพัฒนาเสร็จสิ้น เราจะต้องทำการโหลดข้อมูลเข้าสู่ฐานข้อมูลที่จะสามารถดำเนินการผ่าน 1) โปรแกรมที่ผู้ใช้งานระบบใช้งานอยู่, 2) อินเทอร์เฟซของระบบจัดการฐานข้อมูล และ/หรือ 3) โปรแกรมที่ถูกพัฒนาขึ้นเพื่อใช้สำหรับโหลดข้อมูลโดยเฉพาะ จากนั้นระบบจะถูกตรวจสอบจะกระทั่งพร้อมใช้งาน

โดยปกติของขั้นตอนการสร้างและทดสอบระบบจะใช้เวลาถึง 50 – 60 เปอร์เซ็นต์ของเวลาที่ใช้ในการพัฒนาระบบทั้งหมด แต่อย่างไรก็ตาม ถ้าเราทำการประยุกต์ใช้ซอฟต์แวร์สำเร็จรูปในบางส่วนจะทำให้เราสามารถลดเวลาในการสร้างและการทดสอบระบบลงได้ โดยหลังจากทำการทดสอบระบบจนมีความสมบูรณ์แล้ว ทีมผู้สร้างระบบจะต้องจัดทำเอกสารรายละเอียดต่างๆ เกี่ยวกับระบบ คู่มือการใช้งาน รวมถึงการอบรมเพื่อให้ผู้ใช้สามารถใช้งานระบบใหม่ที่เราสร้างขึ้น โดยหลังจากระบบเริ่มมีการใช้งานแล้วเราจะต้องพิจารณาถึงการประเมินการทำงานของระบบและการปรับปรุงสิ่งต่างๆ ที่จะทำให้ระบบสามารถทำงานได้อย่างมีประสิทธิภาพ

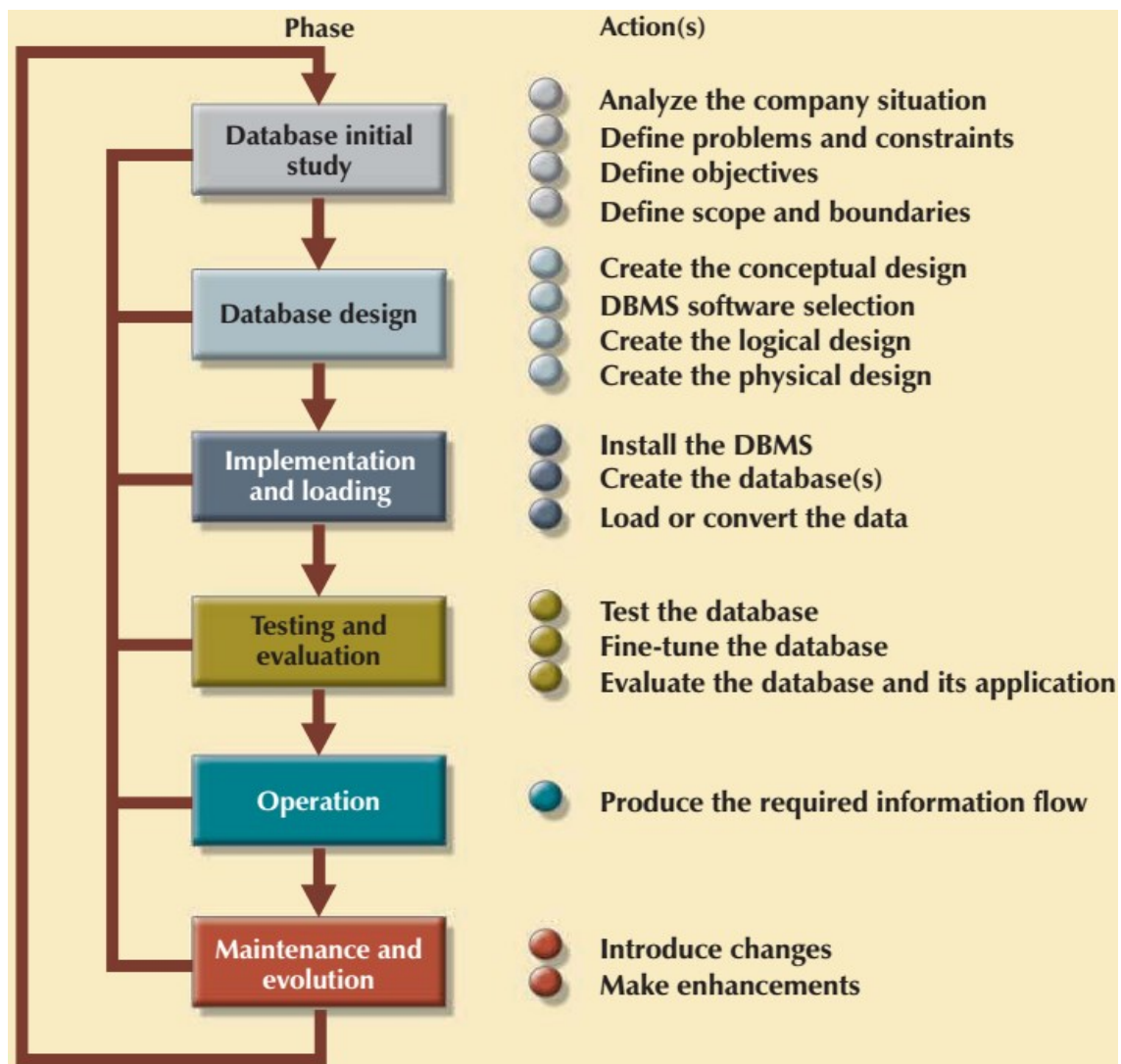
10.2.5 ขั้นตอนการดูแลรักษาระบบ

หลังจากที่ระบบถูกใช้งาน ผู้ใช้มักจะร้องขอสิ่งต่างๆ เพิ่มเติมที่ซึ่งจะก่อให้เกิดกิจกรรมต่างๆ เกี่ยวกับการดูแลรักษาระบบที่ซึ่งสามารถจัดกลุ่มได้เป็น 3 ชนิดหลักดังนี้

- การแก้ไขข้อผิดพลาดของระบบ
- การปรับเปลี่ยนกระบวนการทำงานตามความเปลี่ยนแปลงของการดำเนินธุรกิจ
- การพัฒนาปรับปรุงเพื่อเพิ่มประสิทธิภาพในการทำงานของระบบ

10.3 ขั้นตอนการออกแบบฐานข้อมูล

ภายใต้การออกแบบระบบเทคโนโลยีสารสนเทศ การออกแบบฐานข้อมูลจะเป็นขั้นตอนหนึ่งที่เราจะต้องให้ความสำคัญ ขั้นตอนการออกแบบฐานข้อมูลจะประกอบไปด้วย 6 ขั้นตอนหลักดังแสดงในรูป 10.3 ที่จะประกอบด้วยการศึกษาฐานข้อมูลขององค์กรเบื้องต้น (database initial study), การออกแบบฐานข้อมูล (database design), การสร้างฐานข้อมูลและการโหลดข้อมูลเข้าสู่ฐานข้อมูล (implementation and loading), การทดสอบและประเมินการทำงานของฐานข้อมูล (testing and evaluation), การประยุกต์ใช้ฐานข้อมูล (operation), และการดูแลรักษาและพัฒนาให้ดียิ่งขึ้น (maintenance and evolution) ตามลำดับ

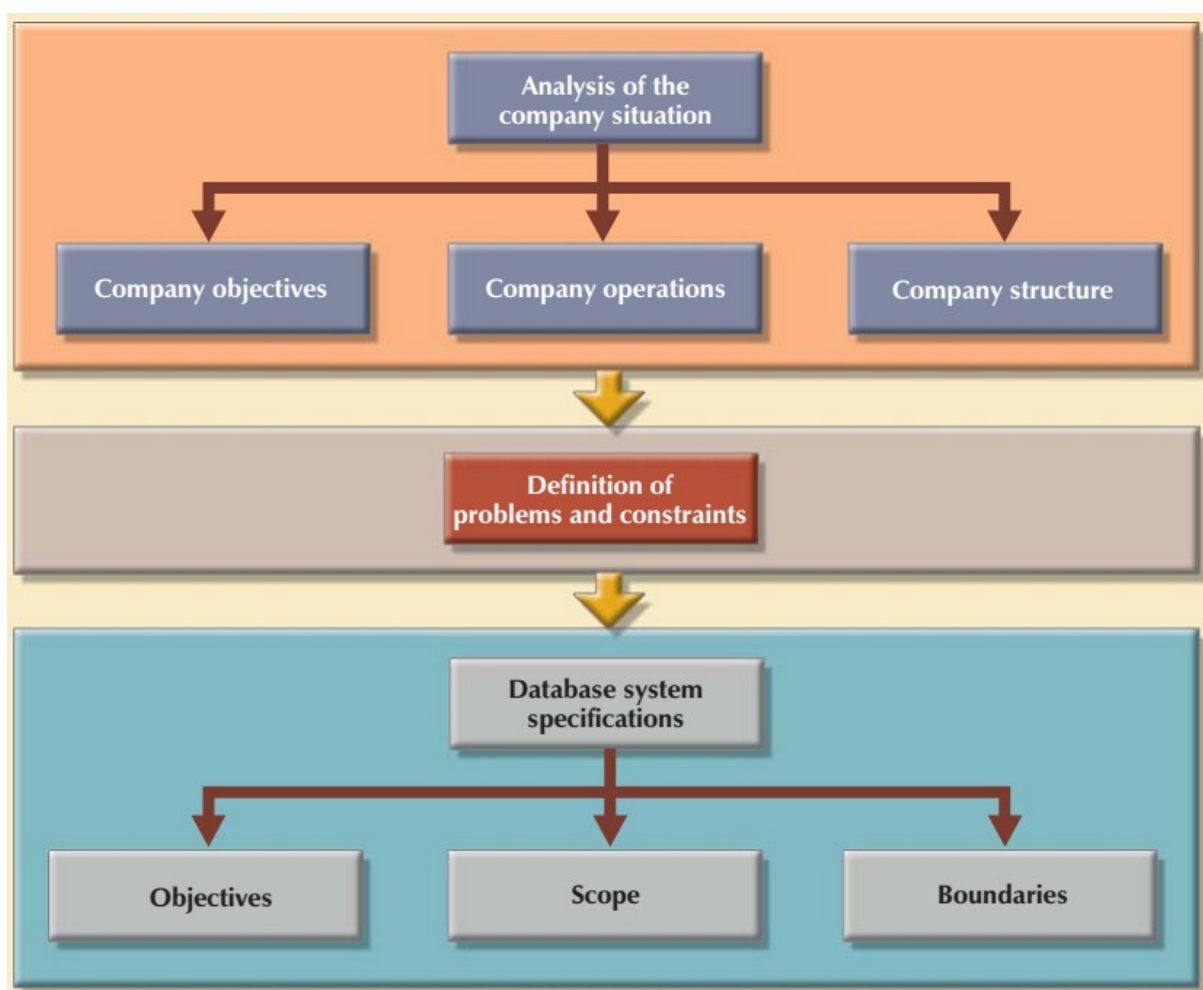


รูปที่ 10.3 ขั้นตอนการออกแบบฐานข้อมูล

10.3.1 การศึกษาฐานข้อมูลขององค์กรเบื้องต้น

ในการศึกษาเกี่ยวกับฐานข้อมูลขององค์กรหนึ่งๆเพื่อทำการปรับปรุงหรือสร้างระบบเทคโนโลยีสารสนเทศขึ้นใหม่ ผู้ออกแบบฐานข้อมูล/ผู้ออกแบบระบบจะต้องเข้าใจถึงสาเหตุและเหตุผลที่ระบบที่ใช้อยู่ในปัจจุบันเกิดความล้มเหลวที่ซึ่งจะสามารถทำความเข้าใจได้ การพูดคุย สัมภาษณ์ และ/หรือสอบถามจากผู้ใช้งานระบบ การศึกษาดังกล่าวจะมีวัตถุประสงค์หลัก 3 ส่วนดังนี้ (ดังแสดงในรูป 10.4)

- การวิเคราะห์สถานการณ์ของบริษัท (analysis of the company situation)
- การระบุถึงปัญหาและเงื่อนไขต่างๆ (define problems and constraints)
- การระบุถึงวัตถุประสงค์ และขอบเขต (Define objectives, scopes and boundaries)



รูปที่ 10.4 สรุปเกี่ยวกับขั้นตอนการศึกษาฐานข้อมูลขององค์กรเบื้องต้น

จากการศึกษาทั้ง 3 ส่วนข้างต้นจะทำให้เราทราบถึงวัตถุประสงค์ของการพัฒนาระบบฐานข้อมูล โดยแต่ละส่วนจะมีรายละเอียดดังนี้

การวิเคราะห์สถานการณ์ของบริษัท

สถานการณ์ของบริษัทจะบ่งบอกถึงการดำเนินการของบริษัท โครงสร้างขององค์กร และเป้าหมายต่างๆขององค์กร ด้วยเหตุนี้จึงเป็นเหตุให้ผู้ออกแบบฐานข้อมูลจะต้องพิจารณาถึงการดำเนินการพื้นฐานและเป้าหมายที่ต้องดำเนินการ และยังรวมถึงการพิจารณาถึงโครงสร้างหลักขององค์กรที่ซึ่งจะทำให้เราทราบถึงลำดับขั้นของพนักงาน และความเกี่ยวเนื่องกันของพนักงานจากสายงานต่างๆ ซึ่งการทราบถึงโครงสร้างขององค์กรจะมีประโยชน์ในการที่จะระบุถึงความต้องการต่างๆของตำแหน่งงานต่างๆ ความต้องการรายงานและคิวรีต่างๆที่เกิดจากหน้าที่ในการดำเนินการต่างๆและอื่นๆ

การระบุถึงปัญหาและเงื่อนไข

ถ้าบริษัทที่ต้องการออกแบบฐานข้อมูลนั้นเปิดทำการมาเป็นเวลานานจะทำให้เราทราบถึงฟังก์ชันการทำงานต่างๆของระบบดั้งเดิม ทราบถึงอินพุตที่ถูกป้อนเข้าไปในระบบ ทราบถึงเอกสารหรือรายงานที่ระบบทำการสร้าง และยังทราบถึงวิธีการใช้เอาท์พุตและบุคคลที่ใช้เอาท์พุตอีกด้วย แต่อย่างไรก็ตาม ในการระบุถึงปัญหานั้นจะทำได้ค่อนข้างยากเนื่องจากผู้ใช้งานจะไม่สามารถบ่งชี้ได้ถึงปัญหาที่กว้างๆได้หรืออาจไม่สามารถระบุถึงปัญหาที่แท้จริงในระหว่างการดำเนินการหนึ่งๆได้ ด้วยเหตุนี้จึงทำให้ผู้ออกแบบฐานข้อมูลจะต้องทำการจัดเก็บคำอธิบายถึงปัญหาอย่างกว้างๆ หลังจากทราบถึงปัญหาและคำอธิบายถึงปัญหาในเบื้องต้นแล้ว ผู้ออกแบบฐานข้อมูลจะต้องไตร่ตรองอย่างละเอียดถี่ถ้วนที่ซึ่งจะได้ข้อมูลสำหรับระบุถึงปัญหา เมื่อทราบถึงปัญหาที่แน่ชัดแล้ว เราจะต้องทำการค้นหาคำตอบหรือวิธีการแก้ปัญหาเหล่านั้นที่จะต้องสอดคล้องกับแนวทางหรือวิธีการในการดำเนินธุรกิจ

การวัตถุประสงค์และระบุขอบเขต

หลังจากการระบุถึงปัญหาและเงื่อนไขต่างๆแล้ว ผู้ออกแบบฐานข้อมูลจะต้องทำให้แน่ใจได้ว่าวัตถุประสงค์ของการสร้างระบบฐานข้อมูล/ระบบเทคโนโลยีสารสนเทศมีความสอดคล้องกับความต้องการหรือปัญหาของผู้ใช้งานหรือไม่ ในการที่จะมีความแน่ใจเกี่ยวกับคำถามข้างต้น ผู้ออกแบบฐานข้อมูลควรที่จะต้องพิจารณาและหาคำตอบสำหรับคำถามดังต่อไปนี้

- วัตถุประสงค์เบื้องต้นของระบบที่จะทำการสร้างขึ้นใหม่คืออะไร?
- ระบบที่จะทำการสร้างขึ้นใหม่สามารถเชื่อมต่อกับระบบเดิมที่มีอยู่หรือสามารถเชื่อมต่อกับระบบที่จะมีการสร้างขึ้นในอนาคตได้หรือไม่?
- ระบบที่จะทำการสร้างขึ้นใหม่มีความสามารถในการแบ่งปันข้อมูลให้กับระบบอื่นๆหรือผู้ใช้งานต่างๆได้หรือไม่?

จากคำถามทั้ง 3 ข้างต้นจะทำให้เราเข้าใจถึงวัตถุประสงค์และลักษณะ/คุณลักษณะของระบบที่จะสร้างขึ้นใหม่ที่ซึ่งจะช่วยให้เรามีความแน่ใจมากยิ่งขึ้นเกี่ยวกับวัตถุประสงค์ที่แท้จริงของการสร้างระบบ

ขอบเขตของระบบจะเป็นตัวกำหนดขอบเขตของการออกแบบฐานข้อมูลสำหรับการดำเนินการต่างๆ ด้วยเหตุนี้เราจำเป็นที่จะต้องทราบอย่างแน่ชัดว่าในการออกแบบครั้งหนึ่งๆจะเป็นการออกแบบสำหรับทั้ง

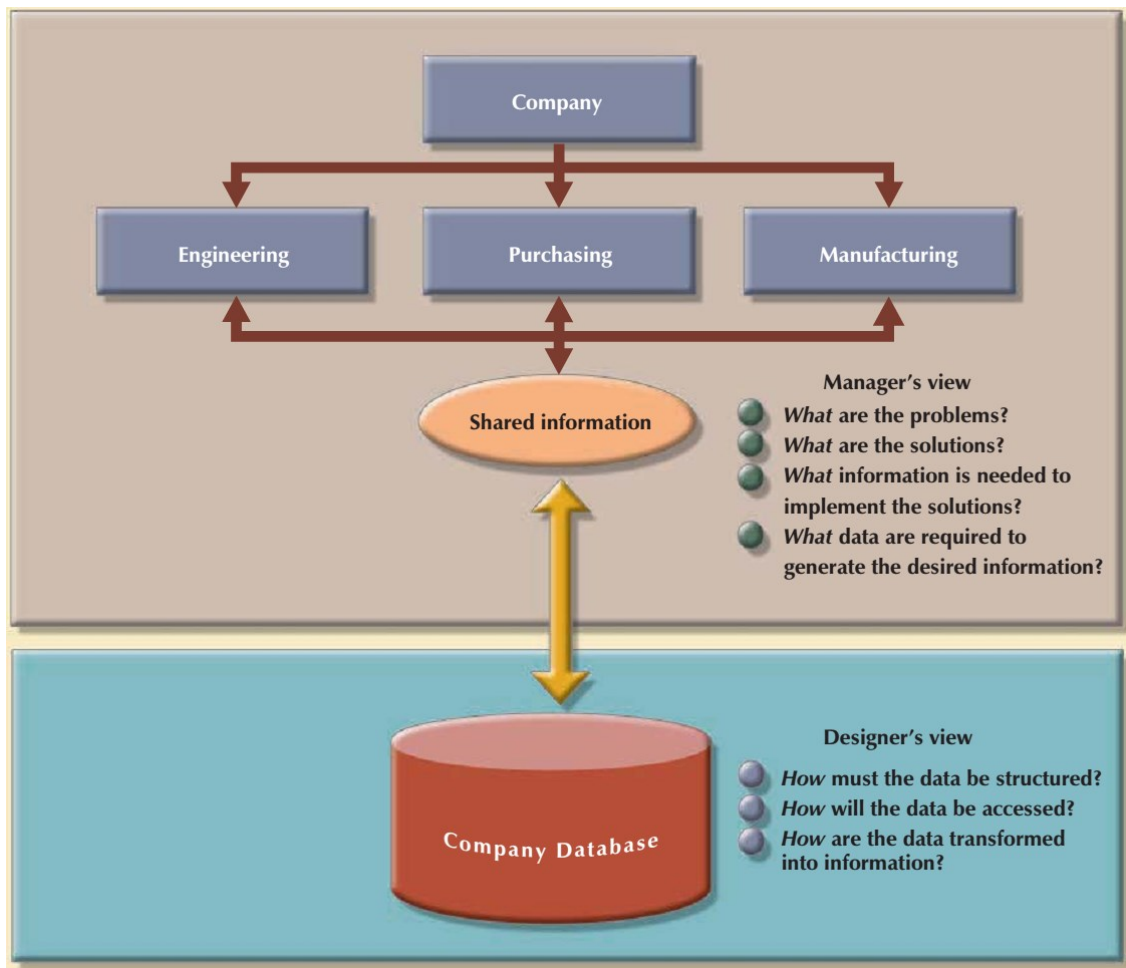
องค์กรหรืออาจเป็นการออกแบบสำหรับแผนกหนึ่งๆหรือหลายๆแผนก หรืออาจเป็นการออกแบบสำหรับฟังก์ชันการทำงานหนึ่งๆภายใต้แผนกหนึ่งๆเท่านั้น การทราบถึงขอบเขตจะช่วยในการกำหนดถึงโครงสร้างข้อมูลที่ต้องการใช้ ชนิดและจำนวนเอ็นทิตีที่ต้องการใช้ และอื่นๆ

ในส่วนขอบเขตอีกขอบเขตหนึ่งจะเกี่ยวข้องกับขอบเขตทางด้านเวลาในการพัฒนาระบบ ขอบเขตทางด้านงบประมาณ และรวมถึงขอบเขตของฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในการสร้างระบบด้วยเช่นกัน ดังนั้นในการศึกษาเกี่ยวกับฐานข้อมูลขององค์กรเราจำเป็นต้องทราบถึงขอบเขตของระบบและเงื่อนไขต่างทั้งในด้านเวลาและงบประมาณที่จะช่วยให้เราตัดสินใจได้ว่าเราจะเลือกใช้ฮาร์ดแวร์ ซอฟต์แวร์และส่วนประกอบอื่นๆอย่างไร

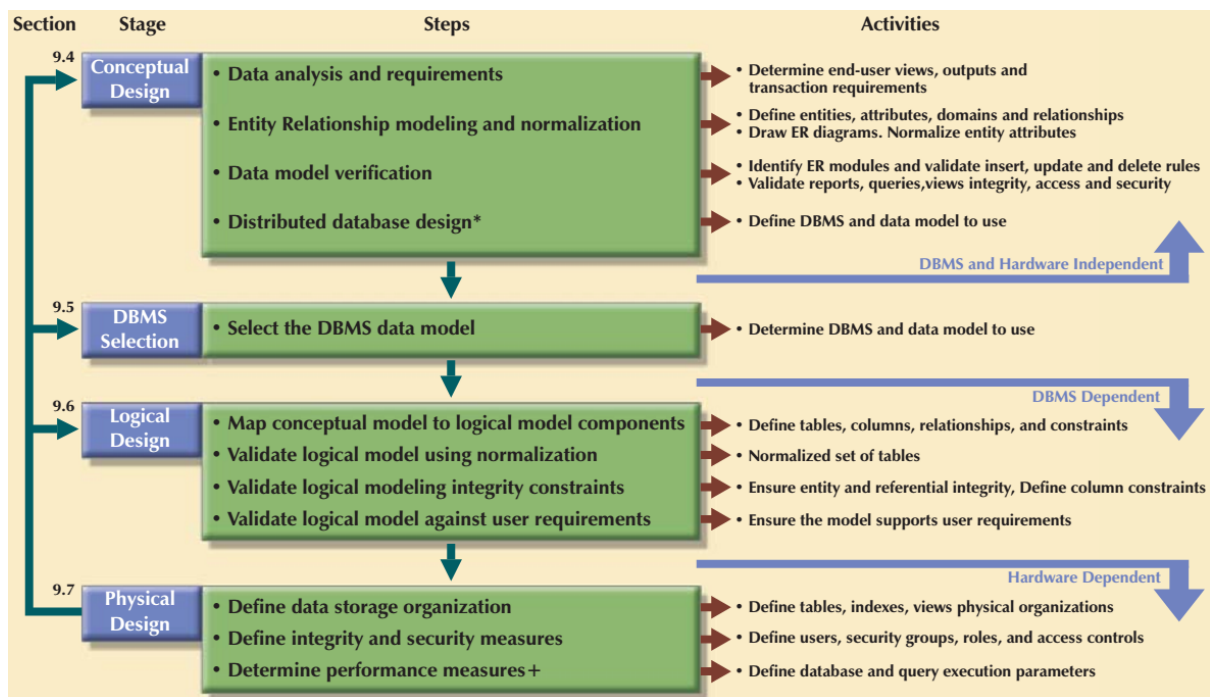
10.3.2 การออกแบบฐานข้อมูล

ขั้นตอนการออกแบบฐานข้อมูลเป็นขั้นตอนสำคัญที่ใช้สำหรับออกแบบฐานข้อมูลเพื่อสนับสนุนการดำเนินการและวัตถุประสงค์ของบริษัท ภายใต้การบวนการออกแบบฐานข้อมูล ผู้ออกแบบต้องให้ความสนใจกับลักษณะ/คุณลักษณะของข้อมูลที่จะทำการจัดเก็บฐานข้อมูล ในการพิจารณาข้อมูลต่างๆจะมีสองมุมมองที่เกี่ยวข้องคือ มุมมองทางธุรกิจที่มองข้อมูลเป็นแหล่งกำเนิดของสารสนเทศ และมุมมองของการออกแบบที่จะพิจารณาโครงสร้างของข้อมูล การเข้าถึง/จัดการกับข้อมูล และกระบวนการต่างๆในการปรับเปลี่ยนข้อมูลให้กลายเป็นสารสนเทศ รูป 10.5 จะแสดงความแตกต่างของมุมมองของ “What” และ “How”

ขั้นตอนการออกแบบฐานข้อมูลจะประกอบไปด้วย 4 ขั้นตอนหลักคือ การออกแบบกรอบความคิด (Conceptual design) , การออกแบบเชิงตรรกะ (Logical design) , การออกแบบเชิงกายภาพ (Physical design) และรวมถึงการเลือกใช้ระบบจัดการฐานข้อมูล (DBMS selection) การออกแบบจะเริ่มจากการออกแบบกรอบความคิดแล้วทำการปรับเปลี่ยนไปเป็นเชิงตรรกะและการออกแบบทางกายภาพ โดยในแต่ละขั้นตอนจะเป็นการเพิ่มรายละเอียดเกี่ยวกับคุณลักษณะของข้อมูลโครงสร้างการจัดเก็บข้อมูล ความสัมพันธ์ต่างๆ และยังรวมถึงการจัดทำเอกสารเพื่อแสดงถึงรายละเอียดต่างๆการออกแบบกรอบความคิดจะเกี่ยวข้องกับข้อมูลทั้งหมดในมุมมองของผู้ใช้งาน (มุมมองทางธุรกิจ) แต่ในส่วนของการออกแบบเชิงตรรกะเกี่ยวข้องกับข้อมูลภายใต้การประยุกต์ใช้ DBMS และท้ายสุดการออกแบบเชิงกายภาพจะเกี่ยวข้องกับข้อมูลที่ถูกดำเนินการโดยระบบปฏิบัติการและอุปกรณ์ฮาร์ดแวร์สำหรับการจัดเก็บข้อมูล (รายละเอียดของขั้นตอนการออกแบบกรอบความคิด , การเลือกใช้ระบบจัดการฐานข้อมูล , การออกแบบเชิงตรรกะ และการออกแบบเชิงกายภาพจะอธิบายรายละเอียดในหัวข้อ 10.4 , 10.5 , 10.6 และ 10.7)



รูปที่ 10.5 ตัวอย่างสองมุมมองของข้อมูล—มุมมองทางธุรกิจและมุมมองทางการออกแบบ



รูปที่ 10.6 ขั้นตอนการออกแบบฐานข้อมูล

10.3.3 การสร้างฐานข้อมูลและการโหลดข้อมูลเข้าสู่ฐานข้อมูล

หลังจากขั้นตอนการออกแบบฐานข้อมูลเราจะทราบถึงตารางข้อมูล (เอ็นทิตี) แอทริบิว ความสัมพันธ์ ขอบเขต มุมมอง ดัชนี และรายละเอียดต่างๆที่เกี่ยวข้องกับการจัดเก็บข้อมูล จากนั้นเราจะทำการดำเนินการสร้างฐานข้อมูลตามทีออกแบบไว้ โดยขั้นตอนการดำเนินการสร้างฐานข้อมูลจะประกอบไปด้วยขั้นตอนย่อย ดังนี้

การติดตั้งระบบจัดการฐานข้อมูล

ในหลายครั้งของการสร้างระบบเทคโนโลยีสารสนเทศ/ระบบฐานข้อมูล ทีมผู้สร้างอาจจำเป็นต้องทำการติดตั้งระบบจัดการฐานข้อมูลในเซิร์ฟเวอร์ใหม่หรือเซิร์ฟเวอร์เดิมที่มีอยู่ แต่ ณ ปัจจุบันทีมผู้สร้างมักจะประยุกต์ใช้เทคนิค “Virtualization” ที่จะทำการสร้างเวิร์ฟเวอร์เสมือน พื้นที่ในการจัดเก็บข้อมูลเสมือน และเครือข่ายเสมือน เพื่อที่จะทำให้การแสดงผลในเชิงตรรกะจะไม่ขึ้นกับฮาร์ดแวร์ต่างๆ หลังจากการสร้างเวิร์ฟเวอร์เสมือนจะทำให้เราสามารถติดตั้งระบบจัดการฐานข้อมูลในเซิร์ฟเวอร์เสมือนได้ (เราจะเรียกฐานข้อมูลที่ถูกติดตั้งในเซิร์ฟเวอร์เสมือนว่า “Database virtualization”)

การสร้างฐานข้อมูล

หลังจากทำการติดตั้งระบบจัดการฐานข้อมูลจะเป็นหน้าที่ของผู้ดูแลระบบ (System administration , SYSADM) และผู้ดูแล (Database administration , DBA) ที่จะทำการสร้างฐานข้อมูล และตารางข้อมูลต่างๆ ในขั้นตอนแรก ผู้ดูแลระบบสามารถทำการสร้างฐานข้อมูล และจากนั้น DBA จะทำการกำหนดสิทธิ์ในการใช้งานฐานข้อมูลให้กับผู้ใช้งาน ทำการสร้างตารางข้อมูลต่างๆภายใต้ฐานข้อมูลที่สร้างขึ้น และทำการกำหนดสิทธิ์ในการเข้าถึง/การจัดเก็บตารางข้อมูลหนึ่งๆด้วยการประยุกต์ใช้คำสั่ง GRANT ดังนี้

GRANT action ON tablename TO username ; ตัวอย่างเช่น ถ้าเราต้องการกำหนดสิทธิ์ให้ Shannon Scott สามารถทำการเรียกดูข้อมูลในตาราง PROFESSOR ได้นั้น เราจะสามารถประยุกต์ใช้คำสั่ง GRANT ได้เป็น GRANT SELECT ON PROFESSOR TO USER SCOTT;

การโหลดข้อมูลเข้าสู่ฐานข้อมูล

หลังจากทำการสร้างฐานข้อมูลและตารางข้อมูลต่างๆแล้วนั้น เราจะสามารถทำการโหลด (จัดเก็บ) ข้อมูลเข้าสู่ฐานข้อมูลได้ โดยในการโหลดข้อมูล เราอาจทำการเคลื่อนย้ายข้อมูลจากระบบดั้งเดิมไปยังฐานข้อมูลที่สร้างขึ้นใหม่ ซึ่งจะอยู่ในรูปแบบฐานข้อมูลเชิงสัมพันธ์เหมือนกัน แต่ก็อาจจะมีหลายๆกรณีที่เราต้องทำการโหลดข้อมูลจากระบบต่างๆ หลายระบบที่ข้อมูลไม่ได้อยู่ในรูปแบบของฐานข้อมูลเชิงสัมพันธ์ อาทิเช่น ข้อมูลที่อยู่ในรูปแบบของแฟ้มข้อมูลต่างๆ สเปรดชีต หรือแม้แต่ข้อมูลที่ถูกจัดเก็บอยู่ในกระดาษ ด้วยเหตุนี้จึงจำเป็นที่จะต้องทำการสร้างโปรแกรมเพื่อทำการปรับเปลี่ยนข้อมูลที่อยู่ในรูปแบบต่างๆให้กลายเป็นข้อมูลที่อยู่ในรูปแบบของฐานข้อมูลเชิงสัมพันธ์ ซึ่งจะทำให้เราสามารถโหลดข้อมูลเหล่านั้นเข้าสู่ฐานข้อมูลที่สร้างขึ้นได้ แต่อาจมีบางกรณีที่เราต้องทำการกรอกข้อมูลแบบ Manual ซึ่งจะต้องทำการตรวจสอบอย่างละเอียดเพื่อหลีกเลี่ยงข้อผิดพลาดในการโหลดข้อมูล

10.3.4 การทดสอบและประเมินการทำงานของฐานข้อมูล

ในขั้นตอนการออกแบบฐานข้อมูล เราต้องทำการตรวจสอบเกี่ยวกับ integrity (entity integrity และ referential integrity), ความปลอดภัย, ประสิทธิภาพ และความสามารถในการกู้คืนข้อมูลของฐานข้อมูลที่ถูกออกแบบ โดยขั้นตอนการตรวจสอบจะประกอบไปด้วยขั้นตอนดังต่อไปนี้

การตรวจสอบฐานข้อมูล

การตรวจสอบฐานข้อมูลจะทำให้เราสามารถแน่ใจได้ว่า 1) ตารางข้อมูลต่างๆมีการกำหนด primary key และ foreign key อย่างเหมาะสม, 2) ตารางข้อมูลต่างๆมีการกำหนดขอบเขตของแอทริบิวต์ ขอบเขตของเงื่อนไข และ database trigger ได้อย่างเหมาะสม และ 3) มีการกำหนดมาตรการความปลอดภัยที่เกี่ยวข้องกับการกำหนดสิทธิ์ในการเข้าถึง/ใช้งานข้อมูลในตารางข้อมูลต่างๆ ตามลำดับ

ในการที่จะทดสอบความปลอดภัยของฐานข้อมูล เราควรที่จะต้องทดสอบในแง่มุมเหล่านี้เป็นอย่างดี:

- Physical security—จะเป็นการยินยอมให้ผู้ใช้ที่มีการยืนยันตัวตนสามารถเข้าใช้งานข้อมูล
- Password security—จะบังคับให้ผู้ใช้งานทำการ log in เพื่อใช้งานข้อมูลตามที่ถูกกำหนดสิทธิ์
- Access rights—จะเป็นการกำหนดสิทธิ์ในการใช้งานข้อมูลในตารางข้อมูลต่างๆเช่น CREATE, UPDATE, DELETE, INSERT, SELECT และอื่นๆ
- Audit trails—จะเป็นการตรวจสอบการล่องละเมิดสิทธิ์ในการดำเนินการกับข้อมูลที่จะดำเนินการโดย DBMS

การประเมินการทำงานของฐานข้อมูลและโปรแกรมต่างๆที่เกี่ยวข้อง

การประเมินประสิทธิภาพการทำงานของฐานข้อมูลจะได้ค่อนข้างยากเนื่องจากการไม่มีมาตรฐานในการประเมิน แต่อย่างไรก็ตามการประเมินประสิทธิภาพของฐานข้อมูลก็เป็นสิ่งจำเป็นอย่างยิ่งที่ซึ่งจะทำให้เราทราบได้ว่าฐานข้อมูลมีการทำงานอย่างไร ระบบฐานข้อมูลที่แตกต่างกันอาจต้องการประสิทธิภาพของการทำงานในแง่มุมที่แตกต่างกัน ตัวอย่างเช่น บางระบบอาจต้องการให้ฐานข้อมูลสามารถเพิ่ม อัปเดต และลบข้อมูลปริมาณมากๆได้อย่างรวดเร็ว หรือ บางระบบอาจต้องการให้ระบบที่สร้างขึ้นสามารถทำการค้นคืนข้อมูลที่มีความซับซ้อนได้อย่างรวดเร็ว และอื่นๆ

จากความต้องการข้างต้นเราจำเป็นอย่างยิ่งที่จะทำการประเมินการทำงานของฐานข้อมูลและโปรแกรมที่เกี่ยวข้องกับฐานข้อมูลที่จะต้องทำการทดสอบประเมินส่วนประกอบย่อยต่างๆของระบบเพื่อให้แน่ใจว่าส่วนต่างๆของระบบสามารถทำงานได้อย่างเหมาะสมและสามารถตอบสนองต่อความต้องการของผู้ใช้งานได้นอกจากนั้นเรายังต้องทำการตรวจสอบเกี่ยวกับกระบวนการสำรองและกู้คืนข้อมูลที่จะสามารถดำเนินการได้หลายระดับดังนี้

- Full backup—จะเป็นการสำรองข้อมูลทั้งหมดในฐานข้อมูล

- Differential backup—จะทำการคัดลอกข้อมูลที่มีการเปลี่ยนแปลงล่าสุดจากการสำรองข้อมูลครั้งก่อนหน้า
- Transaction log backup—จะเป็นการสำรองข้อมูลที่มีการดำเนินการที่มีการจดบันทึกใน log file เท่านั้น

จากการสำรองข้อมูลทั้ง 3 ระดับข้างต้นจะช่วยให้เราสามารถป้องกันอันตรายที่อาจเกิดจากไฟไหม้ ฆโมย น้ำท่วมและอุบัติเหตุหรืออุบัติเหตุต่างๆได้ การสำรองข้อมูลจะช่วยให้องค์กรมั่นใจเกี่ยวกับการกู้คืนข้อมูลเมื่อประสบปัญหาทางด้านซอฟต์แวร์และฮาร์ดแวร์

10.3.5 การประยุกต์ใช้ฐานข้อมูล

หลังจากทำการประเมินและตรวจสอบการทำงานของฐานข้อมูลอย่างแน่ชัดแล้วจะเป็นการประยุกต์ใช้ฐานข้อมูลเพื่อการดำเนินการต่างๆของบริษัท/องค์กร โดยจะเริ่มจากการกำหนดให้ใช้งาน (บางตำแหน่ง/หน้าที่) ให้เริ่มใช้งานฐานข้อมูล เมื่อผู้ใช้งานเริ่มใช้งานจะทำให้เราอาจค้นพบปัญหาที่เราไม่เคยพบมาก่อนในขั้นตอนการทดสอบและตรวจสอบการทำงานของฐานข้อมูล และเมื่อเราค้นพบปัญหาแล้วเราจะต้องทำการระบุถึงวิธีการแก้ไขปัญหาในหลายๆแง่มุมที่ซึ่งจะช่วยให้เราแก้ไขปัญหาได้อย่างสมบูรณ์

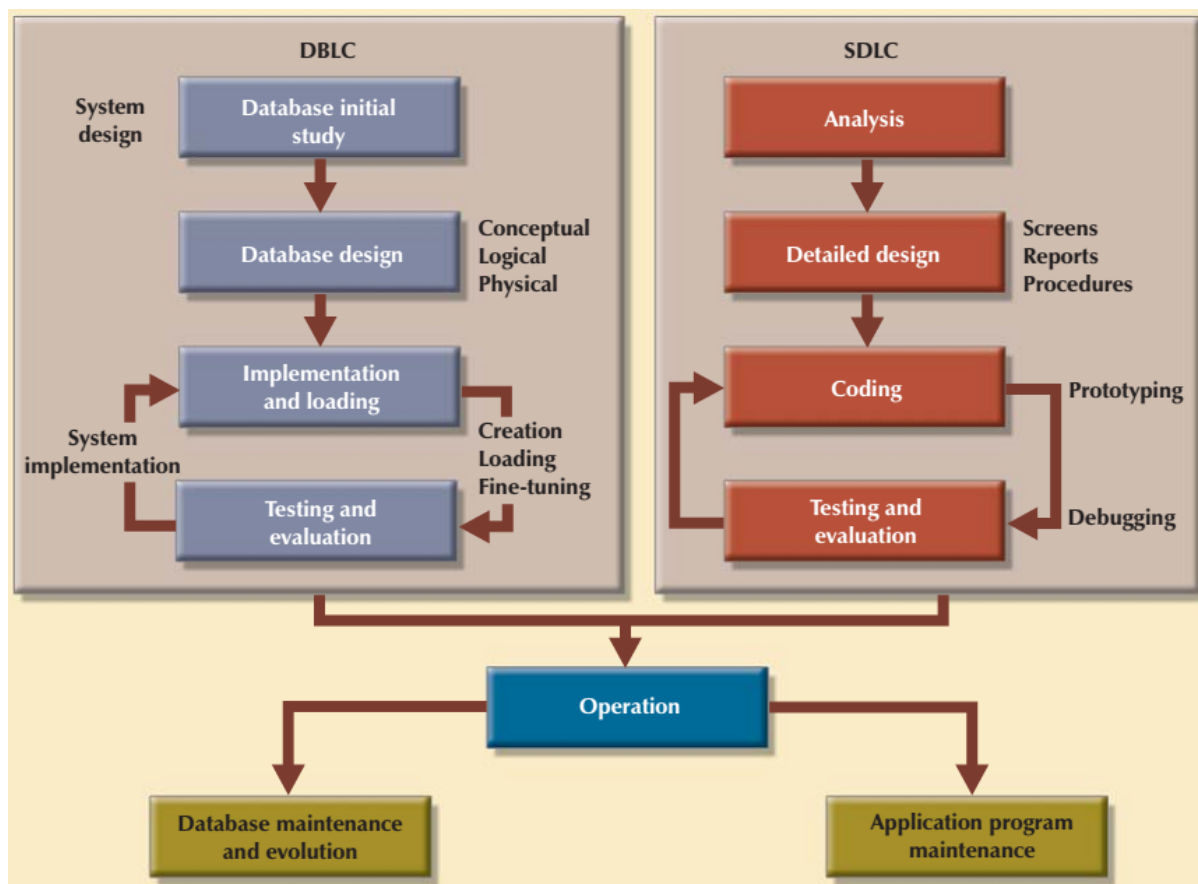
10.3.6 การดูแลรักษาและพัฒนาให้ดียิ่งขึ้น

หลังจากทำการประยุกต์ใช้ฐานข้อมูลแล้ว ผู้ดูแลฐานข้อมูลควรที่จะต้องดำเนินการกิจกรรมต่างๆเพื่อให้ฐานข้อมูลสามารถทำงานได้อย่างราบรื่นดังนี้

- การสำรองข้อมูล
- การกู้คืนข้อมูล
- การเพิ่มประสิทธิภาพการทำงานของฐานข้อมูล
- การเพิ่มเอ็นทิตี แอททริบิว และอื่นๆ
- การกำหนดสิทธิ์ในการใช้งานข้อมูลให้กับผู้ใช้
- การจัดเก็บสถิติเกี่ยวกับการเข้าถึงข้อมูลเพื่อพิจารณาถึงการเพิ่มประสิทธิภาพการทำงานของระบบและเพื่อติดตามเฝ้าสังเกตการทำงานของระบบ
- การตรวจสอบมาตรการความปลอดภัยจากสถิติที่ได้รับ

นอกเหนือจากกิจกรรมทั้งหมดข้างต้น อาจมีความต้องการจากผู้ใช้งานที่จะต้องการข้อมูลสารสนเทศที่เพิ่มขึ้นจากเดิม หรือผู้ใช้อาจร้องขอรายงานต่างๆเพิ่มเติมที่ซึ่งเราจะต้องทำการกำหนดวิธีใหม่ๆเพิ่มเติมด้วยเช่นกัน ถ้าการออกแบบฐานข้อมูลมีความยืดหยุ่นจะทำให้เราสามารถดำเนินการปรับเปลี่ยนระบบเพื่อที่จะสามารถตอบโจทยความต้องการของผู้ใช้งานได้โดยง่าย

จากขั้นตอน DBLC ทั้งหมด เราจะสังเกตได้ว่าขั้นตอนต่างๆใน DBLC จะมีความคล้ายคลึงกับขั้นตอนต่างๆใน SDLC ที่ซึ่งจะสามารถแสดงการทำงานร่วมกันระหว่าง DBLC และ SDLC ได้ดังรูปที่ 10.7



รูปที่ 10.7 การทำงานร่วมกันระหว่างขั้นตอน DBLC และ SDLC

10.4 การออกแบบกรอบความคิด

การออกแบบกรอบความคิดจะเป็นขั้นตอนแรกของการออกแบบฐานข้อมูล (การออกแบบฐานข้อมูลเป็นขั้นตอนที่สองของ DBLC) ที่มีเป้าหมายที่จะออกแบบฐานข้อมูลที่มีความอิสระจากซอฟต์แวร์ฐานข้อมูลและอุปกรณ์ต่างๆที่ใช้สำหรับจัดเก็บข้อมูล โดยผลลัพธ์ที่ได้จากการออกแบบกรอบความคิดจะเป็นแบบจำลองกรอบความคิด (conceptual data model) ที่สามารถบ่งบอกได้ถึงเอ็นทิตี แอททริบิว ความสัมพันธ์และเงื่อนไขต่างๆ การสร้างแบบจำลองกรอบความคิดมักถูกใช้ในการกำหนดโครงสร้างของการจัดเก็บข้อมูลที่จะทำให้มีความเข้าใจในธุรกิจและฟังก์ชันการดำเนินการต่างๆของการดำเนินธุรกิจ ในระหว่างการดำเนินการสร้างแบบจำลองกรอบความคิดเราจะทำการระบุถึงข้อมูลทั้งหมดที่จะถูกใช้ในการดำเนินการต่างๆของการดำเนินธุรกิจ แต่เราไม่ต้องระบุถึงชนิดของฮาร์ดแวร์และซอฟต์แวร์ที่จะใช้สำหรับจัดเก็บและจัดการกับข้อมูล (หมายเหตุ—ด้วยเหตุนี้จึงทำให้การออกแบบแบบจำลองกรอบความคิดจะไม่ขึ้นกับฮาร์ดแวร์และซอฟต์แวร์ที่จะทำให้ระบบที่จะทำการสร้างขึ้นสามารถทำงานได้กับฮาร์ดแวร์และซอฟต์แวร์ใดๆที่จะถูกเลือกใช้ในภายหลัง)

การออกแบบกรอบความคิดจะประกอบไปด้วย 4 ขั้นตอน: การวิเคราะห์ข้อมูลและความต้องการของผู้ใช้งาน, แบบจำลองข้อมูลเชิงสัมพันธ์เอ็นทิตีและกระบวนการนอร์มัลไลเซชัน, การทวนสอบแบบจำลองข้อมูล และการออกแบบฐานข้อมูลแบบกระจาย ที่ซึ่งจะสามารถอธิบายในรายละเอียดดังนี้

10.4.1 การวิเคราะห์ข้อมูลและความต้องการของผู้ใช้งาน

ขั้นตอนแรกของการออกแบบกรอบความคิดคือการระบุถึงลักษณะของข้อมูล โดยข้อมูลเหล่านี้จะถูกปรับเปลี่ยนไปเป็นสารสนเทศเพื่อใช้ในการดำเนินการต่างๆทางธุรกิจ ดังนั้น ผู้ออกแบบควรที่จะมุ่งเน้นที่

- *Information needs*—ผู้ออกแบบจะต้องพิจารณาว่าข้อมูลสารสนเทศใดเป็นที่ต้องการของผู้ใช้และเอาต์พุตของระบบคืออะไร
- *Information users*—ผู้ออกแบบจะต้องพิจารณาว่าใครจะเป็นผู้ใช้ข้อมูลสารสนเทศ ข้อมูลสารสนเทศจะถูกใช้อย่างไร มุมมองข้อมูลของผู้ใช้มีอะไรบ้าง
- *Information sources*—ผู้ออกแบบจะต้องพิจารณาว่าข้อมูลสารสนเทศจะสามารถค้นหาได้จากที่ใด เราจะสามารถสกัดข้อมูลสารสนเทศได้อย่างไร
- *Information constitution*—ผู้ออกแบบจะต้องพิจารณาว่าข้อมูลส่วนใดที่จะถูกประยุกต์ใช้เพื่อสร้างเป็นข้อมูลสารสนเทศ แอทริบิวของข้อมูลมีอะไรบ้าง ความสัมพันธ์ระหว่างข้อมูลมีอะไรบ้าง ปริมาณข้อมูลที่จะถูกจัดเก็บเป็นเช่นไร และข้อมูลจะถูกเข้าถึง/ประยุกต์ใช้บ่อยแค่ไหน

คำถามข้างต้นจะเป็นคำถามที่สำคัญและจำเป็นที่จะต้องค้นหาคำตอบ โดยเราสามารถหาคำตอบได้โดยการดำเนินการดังนี้

- *การพัฒนาและรวบรวมมุมมองข้อมูลของผู้ใช้*—จะช่วยให้เราสามารถระบุได้ถึงข้อมูลต่างๆที่จะถูกจัดเก็บในฐานข้อมูล
- *การสังเกตการทำงานของระบบปัจจุบัน: เอาต์พุตที่มีอยู่และเอาต์พุตที่ต้องการ*--จะทำให้เราทราบถึงรายงานต่างๆที่ผู้ใช้ต้องการและยังสามารถทราบถึงข้อมูลที่สนับสนุนการสร้างรายงานเหล่านั้น
- *การพูดคุยกับทีมผู้พัฒนาระบบ*—อย่างที่เรารับรู้ว่าการออกแบบฐานข้อมูลจะเป็นส่วนหนึ่งของ SDLC ที่ซึ่งผู้ออกแบบระบบก็จะทำหน้าที่ออกแบบกรอบความคิดด้วยเช่นกัน

ในการที่จะพัฒนาแบบจำลองข้อมูลได้อย่างถูกต้อง ผู้ออกแบบควรที่จะต้องมีความเข้าใจเกี่ยวกับชนิดของข้อมูลและการใช้ข้อมูลของบริษัทอย่างถ่องแท้ โดยจากเนื้อหาในบทที่ 2 เราจะทราบได้ว่าถ้าเราสามารถระบุได้ถึงกฎเกณฑ์ทางธุรกิจจะทำให้เราสามารถระบุได้ถึงเอ็นทิตี แอทริบิว ความสัมพันธ์ connectivities cardinalities และ เงื่อนไขต่างๆ เป็นต้น นอกจากประโยชน์ข้างต้นแล้ว การทราบถึงกฎเกณฑ์ทางธุรกิจจะช่วยให้ผู้ออกแบบฐานข้อมูล (แบบจำลองกรอบความคิด) มีความเข้าใจเกี่ยวกับวิธีการในการดำเนินธุรกิจและบทบาทของข้อมูลที่มีส่วนเกี่ยวข้องกับการดำเนินธุรกิจ ดังนั้นเราจะสามารถสรุปประโยชน์ของกฎเกณฑ์ทางธุรกิจที่จะประกอบไปด้วย

- ช่วยในการเป็นตัวกลางในการสื่อสารระหว่างผู้ใช้และผู้ออกแบบ
- ช่วยให้ผู้ออกแบบมีความเข้าใจในธรรมชาติ บทบาท และขอบเขตของข้อมูล
- ช่วยให้ผู้ออกแบบมีความเข้าใจในขั้นตอนการดำเนินธุรกิจ
- ช่วยให้ผู้ออกแบบสามารถพัฒนา/หรือระบุถึงความสัมพันธ์ การมีส่วนร่วมกับความสัมพันธ์ และ foreign key

10.4.2 แบบจำลองข้อมูลเชิงสัมพันธ์เอ็นทิตีและกระบวนการนอร์มัลไลเซชัน

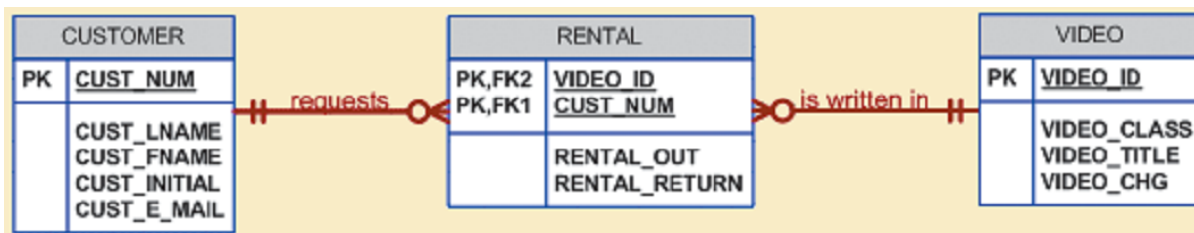
ก่อนที่จะทำการสร้างแบบจำลองเชิงสัมพันธ์เอ็นทิตี ผู้ออกแบบแบบจำลองจะต้องทำการกำหนดมาตรฐานให้กับแผนภาพ สัญลักษณ์ต่างๆ และแนวทางการเขียนเอกสารต่างๆ ที่จะใช้ในการออกแบบแบบจำลองข้อมูล การกำหนดมาตรฐานจะเป็นสิ่งสำคัญมากเมื่อการออกแบบแบบจำลองข้อมูลมีการทำงานเป็นทีมที่ซึ่งจะมีการแบ่งส่วนงานและมีการติดต่อสื่อสารต่างๆที่ซึ่งมาตรฐานจะเป็นตัวกลางให้บุคคลต่างๆในทีมสามารถเข้าใจงานในส่วนต่างๆได้ ดังนั้นถ้าเราไม่ทำการกำหนดมาตรฐานหรือเกิดความล้มเหลวในการกำหนดมาตรฐานแล้วจะทำให้เราเกิดความล้มเหลวในการติดต่อสื่อสารและจะส่งผลให้ได้รับแบบจำลองข้อมูลที่ไม่ดีได้ แต่ในทางกลับกันถ้าเราทำการกำหนดมาตรฐานที่ดีจะช่วยให้การออกแบบนั้นเป็นไปอย่างราบรื่นและยังสามารถรวมส่วนประกอบย่อยต่างๆได้ง่ายอีกด้วย

หลังจากทำการกำหนดมาตรฐานต่างๆที่ใช้ในการออกแบบแบบจำลองข้อมูลแล้ว เราจะสามารถทำการออกแบบจำลองข้อมูลได้โดยประยุกต์ใช้ขั้นตอนดังรูป 10.8 ที่ซึ่งจะเป็นการสร้างแผนภาพเชิงสัมพันธ์เอ็นทิตีที่แสดงถึงโครงสร้างการจัดเก็บข้อมูลในฐานข้อมูล

STEP	ACTIVITY
1	Identify, analyze, and refine the business rules.
2	Identify the main entities, using the results of Step 1.
3	Define the relationships among the entities, using the results of Steps 1 and 2.
4	Define the attributes, primary keys, and foreign keys for each of the entities.
5	Normalize the entities. (Remember that entities are implemented as tables in an RDBMS.)
6	Complete the initial ER diagram.
7	Validate the ER model against the end users' information and processing requirements.
8	Modify the ER model, using the results of Step 7.

รูปที่ 10.8 ขั้นตอนการออกแบบ conceptual model ด้วยการประยุกต์ใช้ ER diagrams

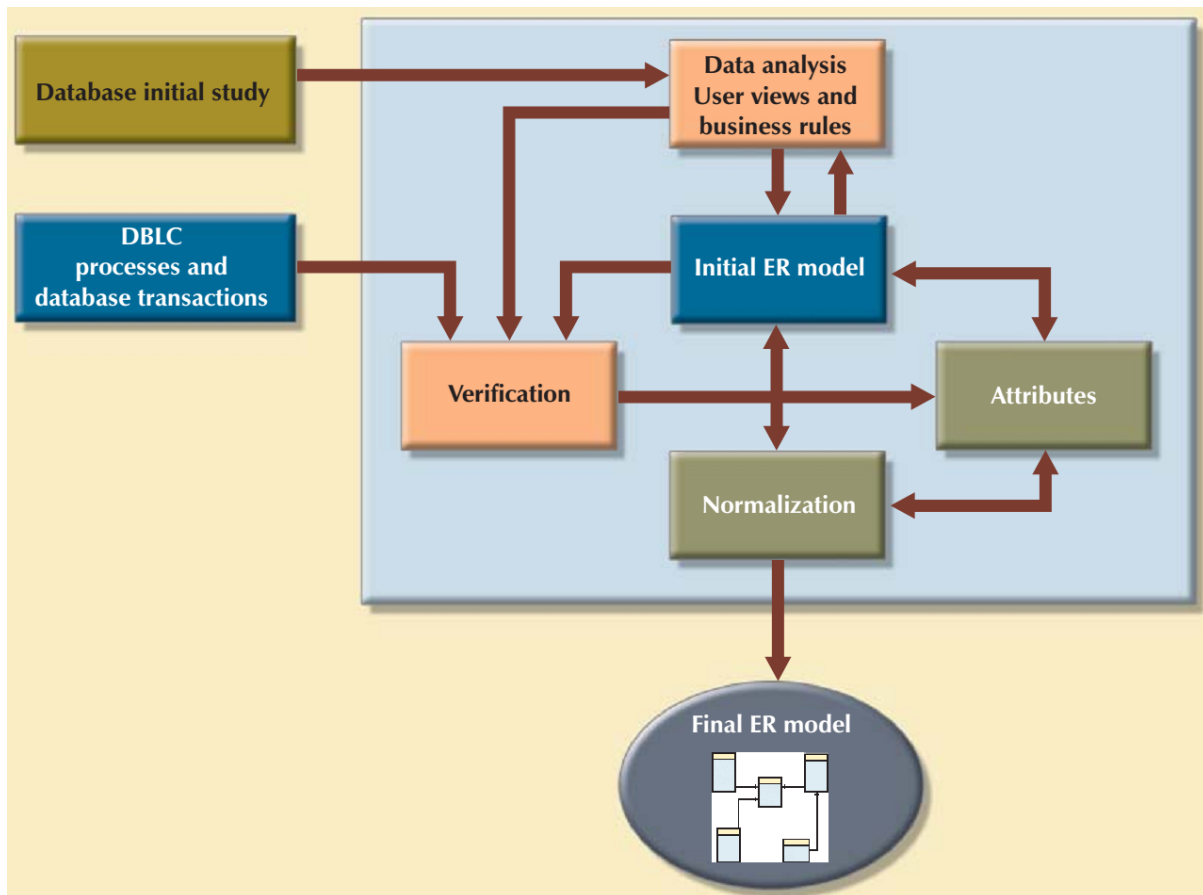
ในการที่จะเข้าใจวิธีการออกแบบ conceptual model ตามขั้นตอนในรูป 10.8 ลองพิจารณาการออกแบบ conceptual model สำหรับบริษัทให้เช่าภาพยนตร์ที่ชื่อ JollyGood Movie Rental Corporation ที่ซึ่งจะต้องการจัดเก็บข้อมูลการเช่าภาพยนตร์ของลูกค้าที่ซึ่งจะสามารถออกแบบ ER diagram อย่างง่ายได้ดังรูป 10.9 ที่เกิดจากกฎเกณฑ์ทางธุรกิจที่ว่า “ลูกค้าคนหนึ่งๆสามารถทำการเช่าภาพยนตร์ได้หลายเรื่อง และภาพยนตร์เรื่องหนึ่งๆสามารถถูกเช่าโดยลูกค้าหลายคน” จากกฎเกณฑ์ทางธุรกิจดังกล่าวจะทำให้เราต้องทำการจัดเก็บข้อมูลลูกค้าในเอ็นทิตี CUSTOMER จัดเก็บข้อมูลภาพยนตร์ในเอ็นทิตี VIDEO และทำการจัดเก็บข้อมูลการเช่าภาพยนตร์ของลูกค้าในเอ็นทิตี RENTAL ที่มีลักษณะเป็น composite entity



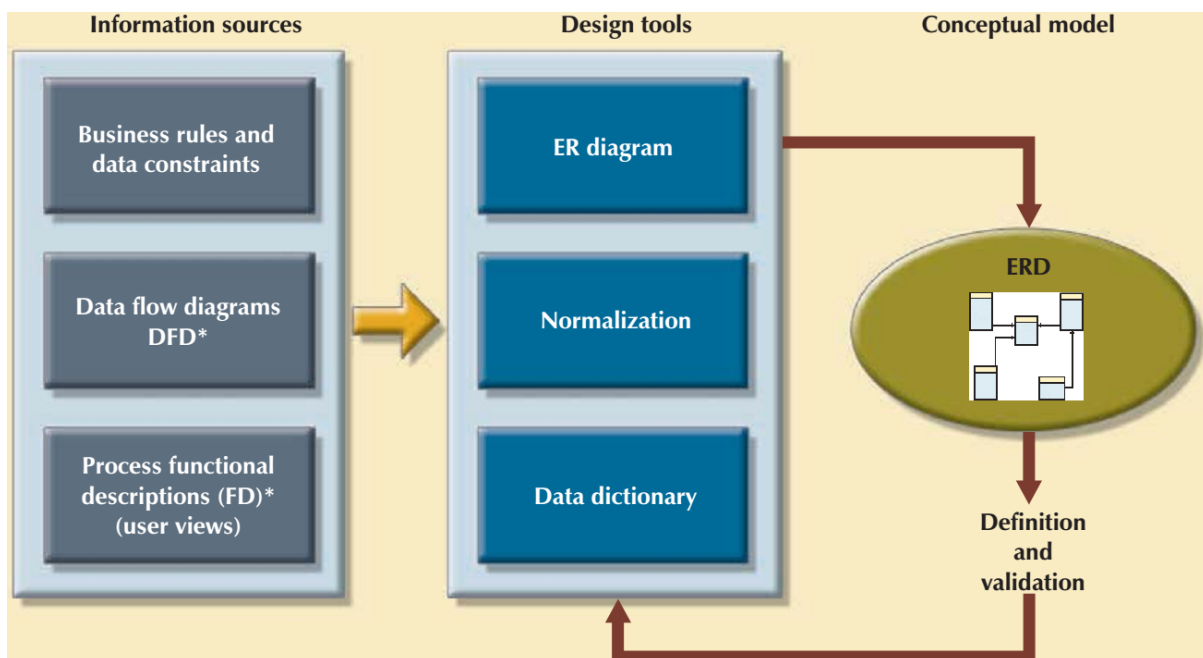
รูปที่ 10.9 ตัวอย่าง ER diagram อย่างง่ายของบริษัทให้เช่าภาพยนตร์

จาก ER diagram ในรูป 10.9 เราควรที่จะตั้งคำถามที่ว่า “โครงสร้างการจัดเก็บข้อมูลใน ER diagram ที่ได้ทำการออกแบบนั้นสามารถจัดเก็บข้อมูลทั้งหมดที่เราต้องการใช้หรือไม่ ?” ซึ่งจาก ER diagram ในรูป 10.9 เราจะสังเกตได้ว่าเรายังกำหนดโครงสร้างในการจัดเก็บข้อมูลไม่ครบถ้วน เราควรที่จะต้องเพิ่มแอททริบิวต์ต่างๆเพื่อใช้ในการจัดเก็บข้อมูลที่สำคัญและเราควรที่จะต้องทำการตรวจสอบการพึ่งพาอาศัย (dependency) กันของแอททริบิวต์ต่างๆ จากขั้นตอนในรูป 10.8 เราจะสังเกตได้ว่าขั้นตอนการออกแบบ ER diagram จะเป็นขั้นตอนการระบุถึงเอ็นทิตี แอททริบิวต์ ความสัมพันธ์ การดำเนินการนอร์มัลไลเซชัน และการตรวจสอบ ER diagram ตามลำดับ โดยในขั้นตอนเริ่มแรกเราจะได้ ER diagram อย่างง่ายที่บ่งบอกถึงโครงสร้างการจัดเก็บข้อมูล จากนั้นเราควรที่จะต้องทำการตรวจสอบซ้ำแล้วซ้ำเล่าเพื่อให้ได้ ER diagram ที่สามารถจัดเก็บข้อมูลที่ผู้ใช้งานต้องการและมีความสอดคล้องกับการดำเนินธุรกิจต่างๆ โดยขั้นตอนการออกแบบ ER diagram จะเป็นกระบวนการทำงานซ้ำๆที่ซึ่งจะเกี่ยวข้องกับกิจกรรมต่างๆดังแสดงในรูป 10.10 และนอกจากนั้นการออกแบบ ER diagram จะต้องยุ่งเกี่ยวเครื่องมือและแหล่งข้อมูลต่างๆดังแสดงในรูป 10.11 จากขั้นตอนในรูป 10.10 จะเกี่ยวข้องกับการกำหนดเอ็นทิตี แอททริบิวต์ ความสัมพันธ์ มุมมองข้อมูล และอื่นๆที่ซึ่งจะมีการจัดเก็บข้อมูลเหล่านี้ในพจนานุกรมข้อมูล จากนั้นจะดำเนินการนอร์มัลไลเซชันเพื่อลดความผิดพลาดและความซ้ำซ้อนของข้อมูล โดยระหว่างขั้นตอนการออกแบบ ER diagram ผู้ออกแบบควรที่จะต้อง

- ระบุถึงเอ็นทิตี แอททริบิวต์ primary keys และ foreign keys
- ตัดสินใจเกี่ยวกับการเพิ่มใหม่ๆเพื่อทำหน้าที่เป็น primary keys ที่จะทำให้โครงสร้างการจัดเก็บข้อมูลสอดคล้องกับการจัดเก็บข้อมูล
- ตัดสินใจเกี่ยวกับการจัดการเกี่ยวกับ composite attribute และ multivalued attribute
- ตัดสินใจเกี่ยวกับการเพิ่ม derived attribute ที่ซึ่งจะสนับสนุนการประมวลผลต่างๆ
- ตัดสินใจเกี่ยวกับการกำหนด foreign key ภายใต้รูปแบบความสัมพันธ์แบบ 1:1
- วาดแผนภาพเชิงสัมพันธ์เอ็นทิตี
- นอร์มัลไลเอ็นทิตีทั้งหมดในแผนภาพเชิงสัมพันธ์เอ็นทิตี
- ระบุถึงนิยามของข้อมูลที่ถูกจัดเก็บในแอททริบิวต์ต่างๆในพจนานุกรมข้อมูล
- ทำการกำหนดมาตรฐานในการตั้งชื่อให้กับแอททริบิวต์และเอ็นทิตีต่างๆหลีกเลี่ยงการปรากฏขึ้นของ ternary relationship



รูปที่ 10.10 ขั้นตอนการออกแบบ ER diagram ที่ประกอบไปด้วยขั้นตอน/กิจกรรมต่างๆ



รูปที่ 10.11 เครื่องมือและแหล่งข้อมูลที่ถูกประยุกต์ใช้ในการออกแบบแบบจำลองข้อมูล

จากขั้นตอนทั้งหมดข้างต้น ขั้นตอนสุดท้ายมักจะถูกละเลยที่ซึ่งจะก่อให้เกิดความสับสนและความเข้าใจผิดเกี่ยวกับข้อมูลได้ ด้วยเหตุนี้ เราควรที่จะต้องให้ความใส่ใจกับการกำหนดมาตรฐานในการตั้งชื่อให้กับแอทริบิวต์และเอนทิตีต่างๆที่จะช่วยให้เราสามารถมีความเข้าใจเกี่ยวกับข้อมูลได้อย่างถูกต้อง

10.4.3 การทดสอบแบบจำลองข้อมูล

การตรวจสอบแบบจำลองข้อมูลจะเป็นขั้นตอนสุดท้ายและเป็นขั้นตอนที่สำคัญในกระบวนการออกแบบแบบจำลองข้อมูล ด้วยเหตุนี้ เราจะต้องทำการตรวจสอบแบบจำลองข้อมูลในหลายๆแง่มุมที่ซึ่งจะประกอบไปด้วยการตรวจสอบดังนี้

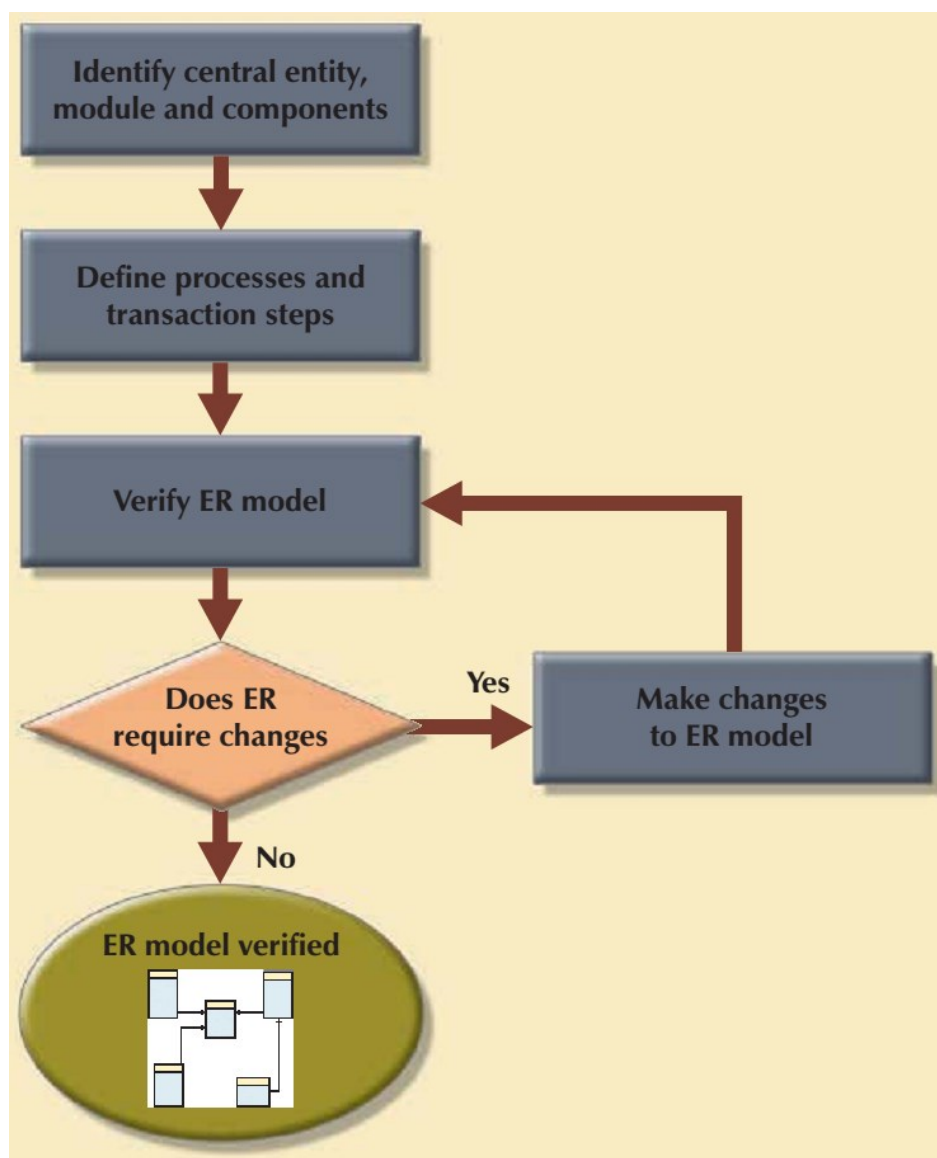
- การตรวจสอบมุมมองข้อมูลของผู้ใช้
- การตรวจสอบการดำเนินการต่างๆกับฐานข้อมูลด้วยการใช้คำสั่งต่างๆเช่น SELECT, INSERT, UPDATE และ DELETE
- การตรวจสอบสิทธิในการใช้งานและมาตรการความปลอดภัย
- การตรวจสอบความสอดคล้องกับการดำเนินธุรกิจและเงื่อนไขต่างๆ

ในการออกแบบฐานข้อมูลส่วนใหญ่จะมีการทำงานเป็นทีมที่จะเป็นกลุ่มผู้ออกแบบที่จะได้รับมอบหมายให้คิดวิเคราะห์ พิจารณาถึงส่วนย่อยหนึ่งของฐานข้อมูล โดยส่วนย่อยหนึ่งๆ (module) จะเป็นส่วนประกอบย่อยหนึ่งๆที่เกี่ยวข้องกับการดำเนินธุรกิจหนึ่งๆ เช่น คลังสินค้า การสั่งซื้อสินค้า การชำระเงิน และอื่นๆ ดังนั้น เมื่อเราต้องทำการแบ่งการออกแบบเป็นส่วนย่อยๆจะช่วยให้เกิดประโยชน์ดังนี้

- ส่วนย่อยต่างๆของระบบจะสามารถแบ่งกระจายให้กับผู้ออกแบบหนึ่งๆที่ซึ่งจะช่วยลดเวลาในการออกแบบฐานข้อมูลได้
- การแบ่งส่วนการออกแบบเป็นส่วนย่อยจะช่วยลดจำนวนเอนทิตีที่ผู้ออกแบบหนึ่งๆต้องพิจารณา อันนำมาซึ่งการลดความผิดพลาดและทำให้ผู้ออกแบบสามารถจัดการออกแบบได้ง่าย
- การแบ่งส่วนการออกแบบเป็นส่วนย่อยจะช่วยให้เราสามารถดำเนินการสร้างระบบฐานข้อมูลต้นแบบได้อย่างรวดเร็ว โดยเมื่อออกแบบส่วนย่อยหนึ่งๆเสร็จเราจะสามารถดำเนินการสร้างระบบฐานข้อมูลได้เลย โดยไม่ต้องรอให้การออกแบบทุกส่วนย่อยเสร็จสิ้น นอกจากนั้นยังช่วยให้ผู้ใช้งานสามารถใช้งานระบบฐานข้อมูลส่วนย่อยที่สร้างเสร็จแล้วได้เร็วขึ้นอีกด้วย

จากแนวทางการออกแบบฐานข้อมูลด้วยการแบ่งเป็นส่วนย่อยจะทำให้แผนภาพเชิงสัมพันธ์เอนทิตีถูกแบ่งเป็นส่วนย่อยด้วยเช่นกัน ด้วยเหตุนี้อาจทำให้แผนภาพเชิงสัมพันธ์เอนทิตีย่อยๆจะเกิดความซ้อนเหลื่อมกันหรืออาจเกิดการมีเอนทิตีที่ซ้ำกัน รวมถึงการมีมุมมองข้อมูลที่ไม่สอดคล้องกัน และ การแบ่งการทำงานออกเป็นส่วนย่อยอาจไม่สนับสนุนการทำงานข้ามส่วนได้ ดังนั้น การที่จะหลีกเลี่ยงปัญหาข้างต้น เราควรที่จะทำการรวบรวมส่วนย่อยต่างๆเข้าด้วยกัน โดยเริ่มจากพิจารณาถึงส่วนย่อยหลักแล้วนำส่วนย่อยต่างๆมารวมกับส่วนย่อยหลักนั้น การรวมแต่ละครั้งจะเป็นการเพิ่มเอนทิตีต่างๆให้กับส่วนย่อยหลัก ด้วยเหตุนี้เราจึงจำเป็นที่

จะต้องทำการตรวจสอบว่าเอนทิตีใหม่ที่ถูกเพิ่มให้กับส่วนย่อยหลักนั้นมีความซ้ำซ้อนกับเอนทิตีเดิมในส่วนย่อยหลักหรือไม่ โดยขั้นตอนการตรวจสอบและการรวมส่วนย่อยๆต่างเพื่อสร้างเป็นแผนภาพเชิงสัมพันธ์เอนทิตีของกระบวนการทั้งหมดจะแสดงในรูป 10.12 ที่ซึ่งจะเป็นกระบวนการทำซ้ำเพื่อตรวจสอบความถูกต้องอย่างละเอียดถี่ถ้วน



รูปที่ 10.12 กระบวนการในการตรวจสอบแบบจำลองข้อมูลเชิงสัมพันธ์เอนทิตี

10.5 การเลือกซอฟต์แวร์ DBMS

การเลือกซอฟต์แวร์ระบบจัดการฐานข้อมูลเป็นกระบวนการหนึ่งที่สำคัญที่ซึ่งจะส่งผลกระทบต่อประสิทธิภาพการทำงานของฐานข้อมูล ดังนั้น ในการที่จะเลือกซอฟต์แวร์ระบบจัดการฐานข้อมูลหนึ่งๆ เราควรที่จะต้องทำการศึกษาค้นคว้าถึงข้อดี-เสียของซอฟต์แวร์นั้นๆ อย่างละเอียดถี่ถ้วนที่ซึ่งจะต้องพิจารณาปัจจัยต่างๆ ดังนี้

- **ค่าใช้จ่าย**—จะรวมถึงค่าใช้จ่ายต่างๆ เช่น ค่าลิขสิทธิ์ซอฟต์แวร์ ค่าดำเนินการติดตั้งซอฟต์แวร์ ค่าใช้จ่ายสำหรับอบรมพนักงานสำหรับใช้งาน และค่าใช้จ่ายในการดำเนินการอื่นๆ
- **คุณลักษณะและเครื่องมือต่างๆของระบบจัดการฐานข้อมูล**—ซอฟต์แวร์ระบบจัดการฐานข้อมูลต่างๆ จะมีฟังก์ชันและเครื่องมือต่างๆเป็นส่วนประกอบ ตัวอย่างเช่น query by example (QBE) การจัดการหน้าจอ (screen management) การสร้างรายงาน การสร้างแอปพลิเคชัน การจัดการพจนานุกรมข้อมูล และอื่นๆ นอกเหนือจากฟังก์ชันข้างต้น เราควรที่จะพิจารณาถึงการดูแล/บำรุงรักษา การประมวลผลคิวรี ความยาก-ง่ายในการใช้งาน ประสิทธิภาพในการทำงาน มาตรการความปลอดภัย และอื่นๆ
- **แบบจำลองข้อมูลที่สอดคล้องกับซอฟต์แวร์ระบบจัดการฐานข้อมูล**—ซอฟต์แวร์หนึ่งๆอาจสำหรับแบบจำลองข้อมูลที่ไม่เหมือนกัน เช่น ซอฟต์แวร์หนึ่งๆอาจสนับสนุนการทำงานภายใต้แบบจำลองข้อมูลแบบลำดับชั้น แบบจำลองข้อมูลแบบเครือข่าย แบบจำลองข้อมูลเชิงสัมพันธ์ หรือแบบจำลองข้อมูลเชิงวัตถุ
- **ความสามารถในการเคลื่อนย้าย**—เราควรที่จะต้องพิจารณาว่าซอฟต์แวร์ DBMS ที่จะเลือกใช้สามารถเคลื่อนย้ายข้ามแพลตฟอร์ม ระบบ และภาษาโปรแกรมได้หรือไม่?
- **ความต้องการฮาร์ดแวร์เพื่อสนับสนุนการทำงานของซอฟต์แวร์ฐานข้อมูล**—จะรวมถึงโปรเซสเซอร์ หน่วยความจำ พื้นที่ในดิสก์ และอื่นๆ

10.6 การออกแบบเชิงตรรกะ

การออกแบบเชิงตรรกะเป็นขั้นตอนที่สองของขั้นตอนการออกแบบฐานข้อมูลที่มีเป้าหมายที่จะทำการออกแบบฐานข้อมูลด้วยการประยุกต์ใช้แบบจำลองข้อมูลหนึ่งๆ การออกแบบเชิงตรรกะจะเป็นขั้นตอนที่ต่อยอดมาจากขั้นตอนการออกแบบกรอบความคิดที่ซึ่งจะนำข้อมูลต่างๆภายใต้กรอบความคิดมากำหนดโครงสร้างการจัดเก็บข้อมูลด้วยการประยุกต์ใช้แบบจำลองข้อมูลหนึ่งๆ ตัวอย่างเช่น ถ้าเราต้องการออกแบบเชิงตรรกะด้วยการประยุกต์ใช้แบบจำลองข้อมูลเชิงสัมพันธ์เราจะต้องทำการระบุถึงตารางข้อมูล แอทริบิวต์ ความสัมพันธ์ และเงื่อนไขต่างๆ เป็นต้น การออกแบบเชิงตรรกะจะประกอบไปด้วย 4 ขั้นตอนหลักดังนี้

- การปรับเปลี่ยนแบบจำลองกรอบความคิดให้กลายเป็นแบบจำลองเชิงตรรกะ
- การตรวจสอบแบบจำลองเชิงตรรกะด้วยการประยุกต์ใช้ نرمัลไลเซชัน
- การตรวจสอบแบบจำลองเชิงตรรกะด้วยกฎความสมบูรณ์ต่างๆ (integrity constraints)
- การตรวจสอบแบบจำลองเชิงตรรกะกับความต้องการของผู้ใช้

ภายใต้ขั้นตอนการออกแบบทั้ง 4 ขั้นตอน เราไม่จำเป็นต้องทำการออกแบบแบบเรียงลำดับ แต่เราจะต้องดำเนินการซ้ำหลายๆรอบเพื่อให้แน่ใจได้ว่าเราสามารถออกแบบแบบจำลองเชิงตรรกะที่มีความสมบูรณ์และถูกต้องตรงความต้องการของผู้ใช้ โดยรายละเอียดของแต่ละขั้นตอนจะสามารถแสดงได้ดังนี้

การปรับเปลี่ยนแบบจำลองกรอบความคิดให้กลายเป็นแบบจำลองเชิงตรรกะ

ขั้นตอนแรกของการออกแบบจำลองข้อมูลเชิงตรรกะจะเป็นการปรับเปลี่ยนแบบจำลองกรอบความคิดให้กลายเป็นแบบจำลองข้อมูลที่ผู้ออกแบบได้ทำการเลือกไว้ แต่ด้วยเนื่องจากเนื้อหาในวิชานี้ได้เน้นย้ำเกี่ยวกับแบบจำลองข้อมูลเชิงสัมพันธ์และแบบจำลองข้อมูลเชิงสัมพันธ์เอ็นทิตี ดังนั้น เราจะเน้นย้ำที่การการปรับเปลี่ยนแบบจำลองกรอบความคิดให้กลายเป็นแบบจำลองข้อมูลเชิงสัมพันธ์เอ็นทิตีที่ซึ่งจะเป็นการนำข้อมูลจากแบบจำลองกรอบความคิดให้กลายเป็นตารางข้อมูล แอทริบิว ความสัมพันธ์ และเงื่อนไขต่างๆ โดยขั้นตอนการปรับเปลี่ยนแบบจำลองกรอบความคิดให้กลายเป็นเซตของตารางข้อมูลจะประกอบไปด้วยขั้นตอนต่างๆดังนี้

- การพิจารณา strong entity
- การพิจารณาความสัมพันธ์ระหว่างเอ็นทิตี supertype และ subtype
- การพิจารณา weak entity
- การพิจารณา binary relationship
- การพิจารณา higher degree relationship

การตรวจสอบแบบจำลองเชิงตรรกะด้วยการประยุกต์ใช้ نرمัลไลเซชัน

หลังจากทำการปรับเปลี่ยนแบบจำลองกรอบความคิดให้อยู่ในรูปแบบของแบบจำลองข้อมูลเชิงสัมพันธ์เอ็นทิตีแล้ว เราจะได้แผนภาพเชิงสัมพันธ์เอ็นทิตีที่บ่งบอกถึงเอ็นทิตี แอทริบิว ความสัมพันธ์และเงื่อนไขต่างๆ แต่อย่างไรก็ตาม ในเอ็นทิตีหนึ่งอาจมีแอทริบิวที่มีลักษณะเป็น composite และ/หรือ multivalued attribute ที่ซึ่งจะต้องการดำเนินการปรับเปลี่ยนให้เป็น simple attribute และ/หรือ single-valued attribute ตามลำดับ จากนั้นเราควรที่จะพิจารณาถึงการขึ้นแก่กันหรือการพึ่งพาอาศัยกันของแอทริบิวต่างๆในแต่ละเอ็นทิตีด้วยการประยุกต์ใช้ขั้นตอน نرمัลไลเซชัน เพื่อทำการปรับเปลี่ยนแบบจำลองข้อมูลให้อยู่ในรูปแบบ 3NF

การตรวจสอบแบบจำลองเชิงตรรกะด้วยกฎความสมบูรณ์ต่างๆ

หลังจากการปรับเปลี่ยนแบบจำลองกรอบความคิดให้กลายเป็นแบบจำลองข้อมูลเชิงสัมพันธ์เอ็นทิตีแล้วเราจะต้องทำการพิจารณาถึงเงื่อนไขต่างๆ รวมถึงการกำหนดขอบเขตค่าของข้อมูลที่ปรากฏในแอทริบิวเพื่อที่จะทำให้ข้อมูลในเอ็นทิตีต่างๆมีความสอดคล้องกับการดำเนินธุรกิจ นอกจากนั้นเรายังจำเป็นต้องทำการพิจารณาเกี่ยวกับ entity integrity และ referential integrity เพื่อที่จะลดความผิดปกติของข้อมูลอีกด้วยท้ายสุดเราต้องทำการตรวจสอบสิทธิ์ในการเข้าถึง/ใช้งานข้อมูลของผู้ใช้งานที่จะช่วยให้ข้อมูลถูกใช้งานอย่างเหมาะสมและมีความปลอดภัย

การตรวจสอบแบบจำลองเชิงตรรกะกับความต้องการของผู้ใช้

ขั้นตอนสุดท้ายของการออกแบบแบบจำลองข้อมูลเชิงตรรกะจะเกี่ยวข้องกับการตรวจสอบแบบจำลองที่ได้ทำการออกแบบไว้กับความต้องการของผู้ใช้งานที่ซึ่งจะต้องทำการร้องขอให้ผู้ใช้งานทำการ

ตรวจสอบข้อมูล และมาตรการความปลอดภัยด้วยเช่นกัน โดยหลังจากผู้ใช้ทำการตรวจสอบอย่างละเอียดถี่ถ้วนแล้ว เราจะได้แบบจำลองข้อมูลเชิงตรรกะที่แสดงถึงโครงสร้างการจัดเก็บข้อมูลที่จะไม่ขึ้นกับฮาร์ดแวร์ต่างๆ และแบบจำลองข้อมูลนี้จะเป็นอินพุตของการออกแบบเชิงกายภาพต่อไป

10.7 การออกแบบเชิงกายภาพ

การออกแบบเชิงกายภาพจะเป็นขั้นตอนการออกแบบองค์ประกอบการจัดเก็บข้อมูล และลักษณะการเข้าถึง/การใช้งานข้อมูลเพื่อให้แน่ใจได้ว่าฐานข้อมูลที่สร้างขึ้นจะมีความสมบูรณ์ มีความปลอดภัย และมีประสิทธิภาพในการทำงาน การออกแบบเชิงกายภาพจะเป็นขั้นตอนที่ต้องการความสามารถเฉพาะที่ซึ่งจะเกี่ยวข้องเกี่ยวกับฮาร์ดแวร์และจะต้องคำนึงถึงประสิทธิภาพการทำงานของระบบ ขั้นตอนการออกแบบเชิงกายภาพจะประกอบไปด้วย 3 ขั้นตอนหลักดังนี้

- การระบุถึงองค์ประกอบการจัดเก็บข้อมูล
- การประเมินความสมบูรณ์และความปลอดภัย
- การตรวจสอบประสิทธิภาพการทำงาน

การระบุถึงองค์ประกอบการจัดเก็บข้อมูล

ก่อนที่เราจะทำการระบุถึงองค์ประกอบของการจัดเก็บข้อมูล เราควรที่จะต้องพิจารณาถึงปริมาณข้อมูลที่ต้องจัดการและรูปแบบการใช้ข้อมูลจากฐานข้อมูลที่จะมีประโยชน์ดังนี้

- การทราบถึงปริมาณข้อมูลจะช่วยให้เราสามารถคำนวณได้ว่าจะต้องทำการจองพื้นที่ในฐานข้อมูลเท่าใดในการจัดเก็บข้อมูล
- การทราบถึงควาถี่ในการเพิ่ม อัปเดต ค้นหาข้อมูลจะช่วยให้เราทราบถึงรูปแบบการใช้งานข้อมูล

เมื่อเราทราบถึงปริมาณข้อมูลและรูปแบบข้อมูลจะทำให้เราสามารถดำเนินการดังต่อไปนี้ได้

- ทำการตรวจสอบพื้นที่สำหรับจัดเก็บข้อมูลในแต่ละตารางข้อมูลและการจัดองค์ประกอบของตารางข้อมูล
- ระบุถึงดัชนีที่จะใช้ในแต่ละตารางข้อมูล
- ระบุถึงมุมมองและประเภทของมุมมองที่จะใช้ในแต่ละตารางข้อมูล

การประเมินความสมบูรณ์และความปลอดภัย

หลังจากทำการระบุถึงองค์ประกอบของการจัดเก็บข้อมูลให้กับตารางข้อมูล การสร้างดัชนี และการสร้างมุมมองสำหรับตารางข้อมูลหนึ่งๆแล้ว ฐานข้อมูลจะมีความพร้อมให้ผู้ใช้สามารถใช้งานได้ แต่ก่อนที่จะใช้งานผู้ใช้ควรที่จะต้องระบุ/ยืนยันตัวตนเพื่อเข้าใช้งาน ด้วยเหตุนี้จึงเป็นเหตุให้เราต้องทำการพิจารณาถึง 1) การกำหนดให้ผู้ใช้เข้าใช้งานและบทบาทของผู้ใช้งาน และ 2) กำหนดความสามารถในการดำเนินการให้กับผู้ใช้ตามลำดับ

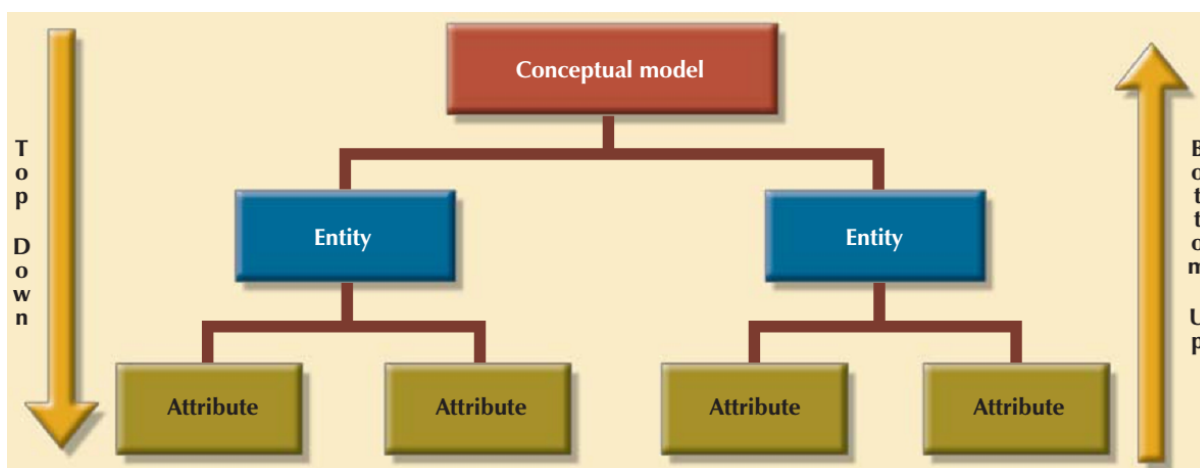
การตรวจสอบประสิทธิภาพการทำงาน

ก่อนที่จะเริ่มใช้งานเราควรที่จะต้องทำการตรวจสอบประสิทธิภาพการดำเนินการต่างๆของฐานข้อมูล ที่ซึ่งจะต้องยุ่งเกี่ยวกับกระบวนการในการปรับฐานข้อมูลและคิวรีต่างๆเพื่อเพิ่มประสิทธิภาพการดำเนินงาน และช่วยให้ประสิทธิภาพการทำงานที่ได้สามารถตอบสนองความต้องการของผู้ใช้งานได้ โดยเนื้อหาในส่วนนี้ เราจะทำการศึกษาอีกครั้งหนึ่งในบทที่ 10

10.8 กลยุทธ์ในการออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลจะมีวิธีดั้งเดิม 2 วิธีดังแสดงในรูป 10.13 ที่จะมีรายละเอียดดังนี้

- Top-down—จะเริ่มจากการระบุถึงข้อมูลอย่างคร่าวๆ แล้วจึงทำการระบุถึงข้อมูลที่มีความละเอียดมากขึ้น ตัวอย่างเช่น เริ่มจากการระบุถึงเอ็นทิตีต่างๆแล้วจึงทำการระบุถึงแอทริบิวต์ต่างๆในเอ็นทิตีเหล่านั้น
- Bottom-up—จะเริ่มจากการระบุถึงข้อมูลที่มีความละเอียดก่อนแล้วค่อยระบุถึงกลุ่ม/ประเภทของข้อมูล ตัวอย่างเช่น การระบุถึงแอทริบิวต์ต่างๆที่จำเป็นต้องทำการจัดเก็บในฐานข้อมูล จากนั้นค่อยทำการรวบรวมแอทริบิวต์ที่เกี่ยวข้องกันเพื่อระบุถึงเอ็นทิตีหนึ่งๆ



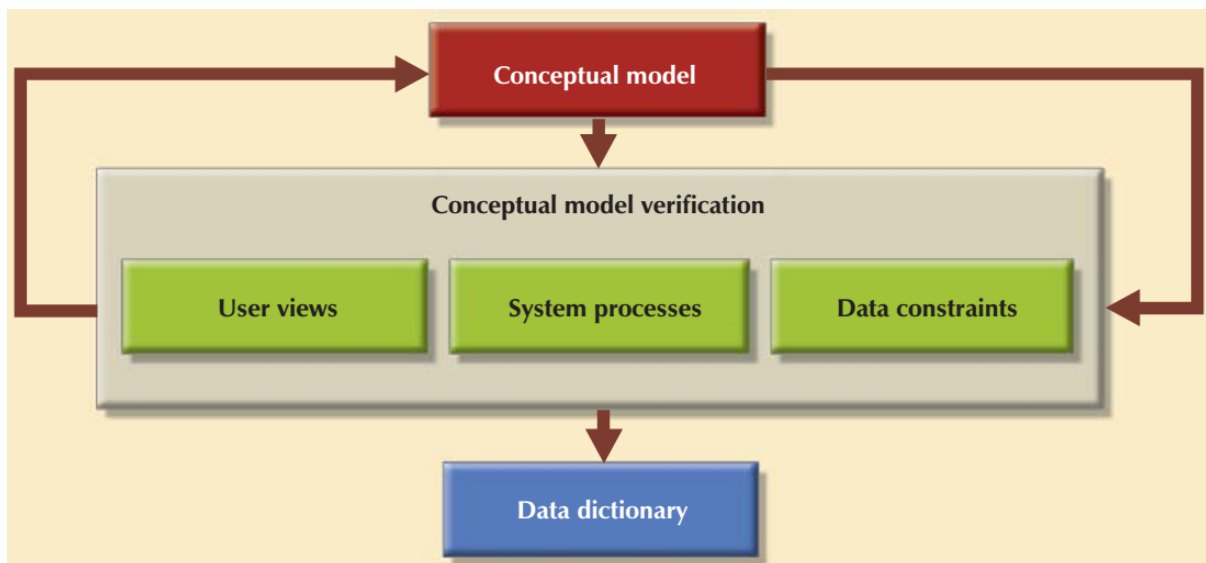
รูปที่ 10.13 วิธีการออกแบบฐานข้อมูลแบบ bottom-up และ top-down

การเลือกวิธีการออกแบบฐานข้อมูลจาก 2 วิธีข้างต้น เราสามารถเลือกได้จากขอบเขตของปัญหาที่จะทำการออกแบบและความชอบหรือความถนัดของผู้ออกแบบ เมื่อเราพิจารณาคร่าวๆ เราอาจมองว่าทั้งสองวิธีนี้คล้ายคลึงกัน แต่โดยแท้จริงแล้วการออกแบบในลักษณะ bottom-up จะมีประสิทธิภาพในการออกแบบฐานข้อมูลขนาดเล็กที่มีเอ็นทิตี แอทริบิวต์ ความสัมพันธ์ และเงื่อนไขไม่มากนัก แต่สำหรับการออกแบบในลักษณะ top-down จะเหมาะสำหรับการออกแบบฐานข้อมูลขนาดใหญ่ที่มีความหลากหลาย ซับซ้อน ที่ซึ่งวิธีการนี้จะช่วยให้เราสามารถบริหารจัดการได้โดยง่าย

10.9 การออกแบบในลักษณะ centralized และ decentralized

การเลือกวิธีการออกแบบฐานข้อมูลแบบ bottom-up หรือ top-down จะขึ้นกับขอบเขตและขนาดของระบบฐานข้อมูลที่จะทำการออกแบบ แต่อย่างไรก็ตาม เราควรที่จะพิจารณาเกี่ยวกับโครงสร้างขององค์กรที่จะมีลักษณะเป็นแบบ centralized หรือ decentralized ที่จะเกี่ยวเนื่องกับหลักการการออกแบบฐานข้อมูล 2 หลักการดังนี้

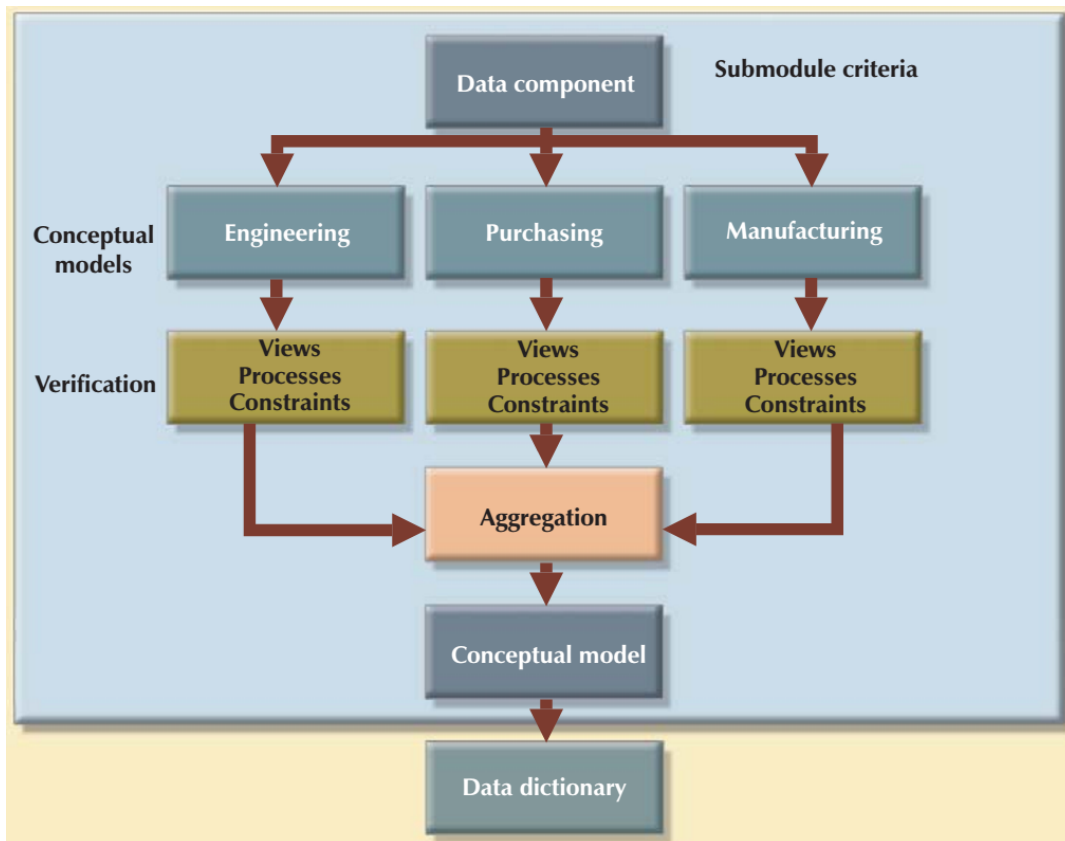
- Centralized design—จะเป็นหลักการออกแบบฐานข้อมูลขนาดเล็กที่มีการดำเนินการกับฐานข้อมูลไม่มากนัก ดังนั้น การออกแบบฐานข้อมูลอาจดำเนินการด้วยผู้ออกแบบเพียงคนเดียวหรืออาจดำเนินการด้วยทีมผู้ออกแบบกลุ่มเล็กๆ การออกแบบแบบ centralized จะประกอบไปด้วยขั้นตอนการออกแบบดังแสดงในรูป 10.14 ที่จะประกอบไปด้วยการระบุถึงปัญหา/การดำเนินการต่างๆ การออกแบบกรอบความคิด การตรวจสอบความคิดที่ออกแบบไว้ การกำหนดฟังก์ชันการทำงาน/การดำเนินการต่างๆของระบบ การกำหนดเงื่อนไขของข้อมูล และการตรวจสอบกรอบความคิดว่ามีความสอดคล้อง/สามารถตอบสนองต่อการดำเนินการทางธุรกิจและสิ่งที่ผู้ใช้ต้องการ ตามลำดับ เมื่อทำการออกแบบตามขั้นตอนข้างต้นจะทำให้เราได้กรอบความคิดและพจนานุกรมข้อมูลที่บ่งบอกถึงโครงสร้างการจัดเก็บข้อมูล



รูปที่ 10.14 ขั้นตอนการออกแบบฐานข้อมูลแบบ centralized design

- Decentralized design—มักถูกประยุกต์ใช้ในกรณีที่ระบบฐานข้อมูลมีเอ็นทิตีที่ต้องพิจารณาเป็นจำนวนมาก เอ็นทิตีที่มีความสัมพันธ์ที่ซับซ้อน และมีการดำเนินการกับฐานข้อมูลที่มีความซับซ้อน นอกจากนั้น decentralized design ยังถูกประยุกต์ใช้กับองค์กรที่มีการดำเนินการในหลายๆพื้นที่ โดยแต่ละพื้นที่จะมีการจัดเก็บข้อมูล (ดังแสดงตัวอย่างดังรูป 10.15) เมื่อระบบฐานข้อมูลมีขนาดใหญ่ และมีความซับซ้อน เราอาจไม่สามารถออกแบบฐานข้อมูลเพียงลำพังได้ ดังนั้นเราควรจะต้องมีการทำงานเป็นทีมที่ซึ่งจะแบ่งการทำงานออกเป็นโมดูลย่อยๆแล้วกำหนดให้แต่ละคนในทีมรับผิดชอบ

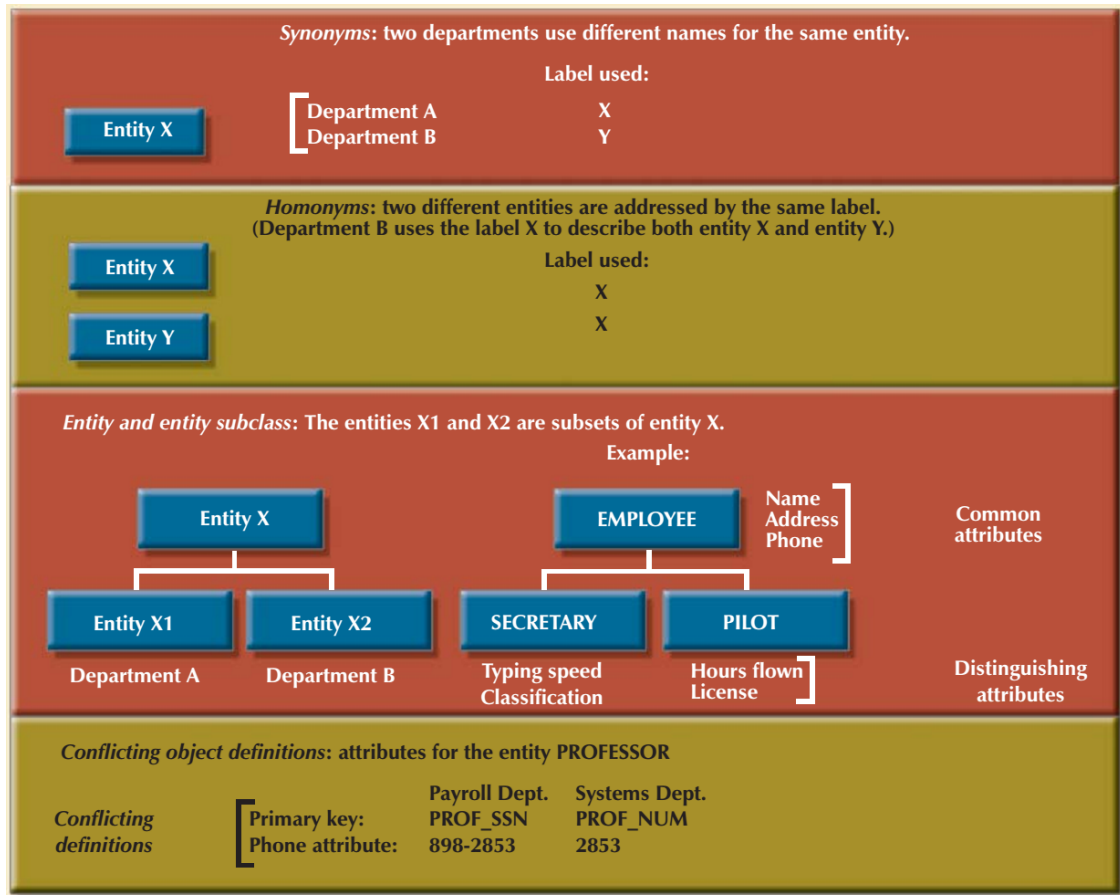
โมเดลนั้นๆที่จะรับผิดชอบการออกแบบบางส่วนของระบบฐานข้อมูลเท่านั้น เมื่อผู้ออกแบบได้รับมอบหมายให้ดำเนินการออกแบบโมเดลหนึ่งๆแล้ว พวกเขาจะต้องทำการออกแบบกรอบความคิด และทำการตรวจสอบกรอบความคิดที่ได้ออกแบบไว้ เมื่อทุกกรอบความคิดย่อยได้ถูกตรวจสอบจนครบถ้วนแล้ว กรอบความคิดเหล่านั้นจะถูกรวมเข้าด้วยกันเพื่อสร้างเป็นกรอบความคิดของทั้งระบบ และท้ายสุดจะเป็นการกำหนดเกี่ยวกับพจนานุกรมข้อมูลสำหรับข้อมูลภายใต้กรอบความคิดทั้งหมดจากขั้นตอนทั้งหมด ขั้นตอนการรวมข้อมูลจะเป็นขั้นตอนที่เราต้องดำเนินการอย่างระมัดระวังที่ซึ่งอาจก่อให้เกิดปัญหาดังนี้ (ดังแสดงในรูป 10.16)



รูปที่ 10.15 ขั้นตอนการออกแบบฐานข้อมูลแบบ decentralized design

- Synonyms และ homonyms--หลายๆแผนกอาจจะไปถึงเอ็นทิตี/แอทริบิวเดียวกันในหลายๆลักษณะ (synonyms) หรือ แผนกต่างๆอาจทำการกำหนดชื่อของข้อมูลหนึ่งๆกับข้อมูลที่มีความแตกต่างกัน (homonyms)
- Entity supertypes และ entity subtypes—เอ็นทิตี subtype หนึ่งๆจากมองได้เป็นเอ็นทิตีย่อยของแผนกหนึ่งๆหรือหลายๆแผนก ด้วยเหตุนี้ผู้ออกแบบฐานข้อมูลจะต้องทำการรวมเอ็นทิตี subtype ให้อยู่ในระดับที่สูงขึ้นใน specialization hierarchy
- Conflicting object definitions—แอทริบิวหนึ่งๆอาจถูกจัดเก็บในหลายๆแผนกที่ซึ่งจะใช้ชนิดข้อมูลที่แตกต่างกัน เช่น แผนกหนึ่งๆอาจจัดเก็บเป็นข้อมูลเชิงตัวเลข แต่อีกแผนกหนึ่ง

อาจจัดเก็บเป็นข้อมูลเชิงตัวอักษร นอกจากนั้นแล้ว แอทริบิวต์หนึ่งๆอาจมีขอบเขตของค่าข้อมูลที่ปรากฏแตกต่างกันได้ ด้วยเหตุนี้ ผู้ออกแบบจะต้องทำการตรวจสอบและปรับเปลี่ยนให้แอทริบิวต์เหล่านั้นมีชนิดและขอบเขตของข้อมูลที่เหมือนกัน



รูปที่ 10.16 ปัญหาของการรวบรวมกรอบความคิดย่อย

คำถามท้ายบท

1. ระบบเทคโนโลยีสารสนเทศคืออะไร? ระบบนี้มีวัตถุประสงค์อย่างไร?
2. SDLC คืออะไร?
3. DBLC คืออะไร?
4. กฎเกณฑ์ทางธุรกิจคืออะไร? และมีความสำคัญอย่างไร?
5. ขั้นตอนการออกแบบ ER diagram ประกอบไปด้วยขั้นตอนอะไรบ้าง?
6. การเลือกซอฟต์แวร์ระบบจัดการฐานข้อมูลมีความสำคัญอย่างไร?
7. ขั้นตอนการออกแบบ logical design ประกอบไปด้วยขั้นตอนอะไรบ้าง?
8. ขั้นตอนการออกแบบ physical design ประกอบไปด้วยขั้นตอนอะไรบ้าง?
9. ความแตกต่างระหว่างการออกแบบฐานข้อมูลแบบ bottom-up และ top-down เป็นอย่างไร?
10. ความแตกต่างระหว่าง centralized และ decentralized conceptual design เป็นอย่างไร?