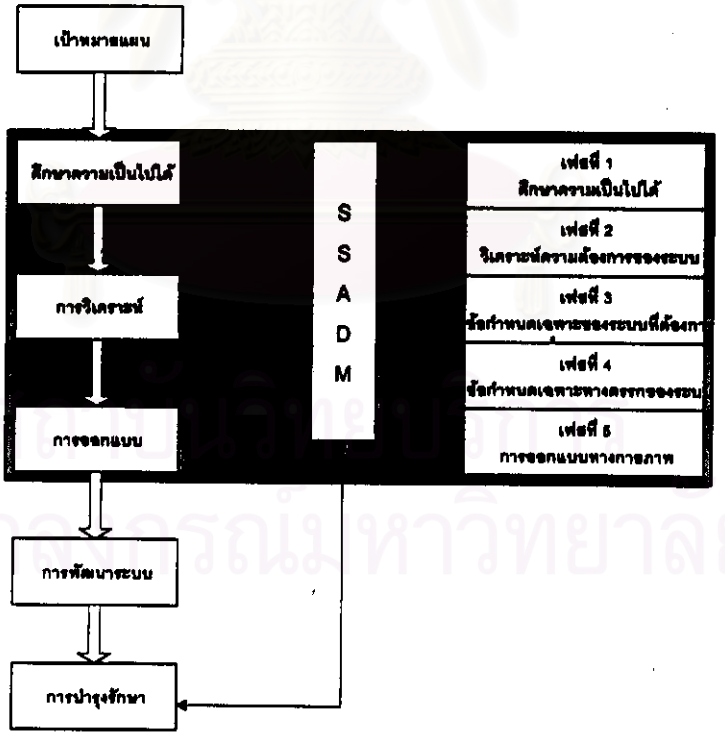
บทที่ 2

วิธีการวิเคราะห์และออกแบบ SSADM

วิธีการวิเคราะห์และออกแบบ SSADM (Structured System Analysis and Design Method) ถูกคิดค้นและพัฒนาโดย Learmonth and Burchette Management System (LBMS) เป็นแนวความคิดวิธีการวิเคราะห์และออกแบบที่ใช้กันอย่างมากที่ประเทศอังกฤษ โดยมีการพัฒนามาหลายรุ่นจนปัจจุบันเป็นเวอร์ชั่น 4 (Philip L. Weaver, 1993) ในขั้นตอนการพัฒนาระบบสารสนเทศหนึ่ง ๆ นั้น ขั้นตอนการวิเคราะห์และออกแบบถือเป็นหัวใจสำคัญ วิธีการวิเคราะห์และออกแบบ SSADM เป็นขั้นตอนที่อยู่ในขั้นตอนที่ 2 ถึงขั้นตอนที่ 4 ของการพัฒนาระบบสารสนเทศ ดังรูป 2.1 SSADM มีมุมในการมองระบบที่แตกต่างกัน 3 มุมมอง ดังนี้ มองที่หน้าที่หรือขั้นตอนการดำเนินงาน (Functionality or Processing View) มองที่ข้อมูล (Data View) เป็นมุมมองที่สำคัญของการวิเคราะห์และออกแบบ SSADM และมุมมองสุดท้ายคือมองที่เหตุการณ์ (Event View) โดยดูที่ผลกระทบของเวลาว่าจะเกิดอะไรขึ้นเมื่อเวลาเปลี่ยนไป



รูป 2.1 แสดงวงจรการทำงานของ SSADM

ส่วนประกอบของ SSADM จะแบ่งได้ 3 ส่วนคือ กรอบขั้นตอนการทำงาน (Structure or Framework) กลุ่มของเครื่องมือ/เทคนิคมาตรฐานที่ใช้ในการวิเคราะห์และออกแบบ และผลลัพธ์เกิดขึ้นในแต่ละเทคนิคที่ใช้ โดยจะได้อธิบายแต่ละส่วนประกอบดังต่อไปนี้

2.1 กรอบขั้นตอนการทำงาน (Structure or framework)

จะแบ่งออกเป็นเฟส และในแต่ละเฟสจะมีสเตจ ของการทำงาน นอกจากนี้แล้วในแต่ละสเตจจะแบ่งการทำงานเป็นเสต็ป และแต่ละเสต็ปจะมีทาสก์ ซึ่งทาสก์ก็คือขั้นตอนวิธีการทำงานของเทคนิคที่ใช้ รายละเอียดของแต่ละขั้นตอนมีดังต่อไปนี้

2.1.1 เฟสที่ 1 ศึกษาความเป็นไปได้ของระบบ (Feasibility Study) ประกอบด้วยสเตจ 0

สเตจ 0 ศึกษาความเป็นไปได้ของระบบ ก่อนที่จะมีการพัฒนาหรือสร้างระบบสารสนเทศ จะต้องมีการศึกษาถึงขั้นตอนการทำงานของระบบปัจจุบันและความต้องการของระบบ โดยคำนึงถึงค่าใช้จ่ายและเวลาที่ใช้ว่าคุ้มกับการที่จะพัฒนาระบบหรือไม่ และเสนอให้ผู้บริหารโครงการทราบเพื่อตัดสินว่าจะมีการพัฒนาต่อหรือไม่ สเตจ 0 ประกอบไปด้วยเสต็ปย่อย ๆ ดังรูปที่ 2.2 และมีรายละเอียดของแต่ละเสต็ปดังต่อไปนี้

รูป 2.2 แสดงกรอบการทำงานของสเตจ 0

1) เสต็ป 010 การเตรียมพร้อมเพื่อศึกษาความเป็นไปได้ของระบบ เพื่อเป็นการประเมินขอบเขตของโครงการและวางแผนการศึกษาของระบบ ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูปที่ 2.3

2) เสต็ป 020 การกำหนดปัญหา เป็นการศึกษาความต้องการของระบบใหม่ เพื่อเป็นการกำหนดแนวทางในการแก้ปัญหา และเพื่อเปรียบเทียบกับระบบเดิม โดยอาศัยผลลัพธ์ที่ได้จากเสต็ป 010 พร้อมกันนี้ยังปรับปรุงแผนภาพระบบงานเดิม ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.4

3) เสต็ป 030 การเลือกทางเลือกที่เป็นไปได้ เป็นการเลือกทางเลือกในการแก้ปัญหาของระบบโดยต้องมีทางเลือกหลายทาง จะประกอบด้วยทางเลือกทางด้านธุรกิจ (Business System Option, BSO) เป็นความต้องการที่ผู้ใช้ต้องการกับทางเลือกทางด้านเทคนิค (Technical System Option, TSO) เป็นแนวทางในการพัฒนาระบบ ซึ่งรวมทั้งคำอธิบายในด้านฮาร์ดแวร์และซอฟท์แวร์ ประกอบด้วยทาสก์ต่าง ๆ เทคนิคและผลลัพธ์ที่ได้ดังรูป 2.5

ทาสก์ : 10 ศึกษาถึงเอกสารและกำหนดขอบเขตของโครงการโดยการสร้างแบบจำลองแสดงขั้นตอนการทำงานและข้อมูลในปัจจุบัน 20 ตรวจสอบขอบเขตของโครงการกับผู้ใช้	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
แบบจำลองกระแสข้อมูล (Data Flow Model, DFM)	- แผนภาพกระแสข้อมูลหลักของระบบงานปัจจุบัน (Context Diagram - Current Environment) - แผนภาพรวมกระแสข้อมูลของระบบงานปัจจุบัน (Overview DFM - Current Environment)
แบบจำลองข้อมูลเชิงตรรก (Logical Data Model, LDM)	- แผนภาพรวมของข้อมูลเชิงตรรกของระบบปัจจุบัน (Overview LDM - Current Environment)
การกำหนดความต้องการ (Requirements Definition)	- รายการความต้องการของระบบ (Requirements Catalogue)

รูป 2.3 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 010

ทาสก์ : 10	สร้างคำอธิบายถึงความต้องการของระบบใหม่โดยการใช้แผนภาพรวมของกระแสข้อมูลระบบงานเดิมกับแผนภาพรวมของข้อมูลตรรกประกอบ
20	ปรับปรุงขยายแผนภาพรวมของกระแสข้อมูลและแผน ภาพรวมข้อมูลตรรกของระบบงานเดิม และสร้างคำอธิบายระบบงานเดิม บันทึกปัญหาต่าง ๆ ของระบบงานเดิมลงในรายการตามความต้องการของระบบ
30	บันทึกความต้องการของระบบในมุมมองของผู้ใช้
40	กำหนด/เพิ่มฟังก์ชันที่ต้องการและไม่ต้องการของระบบใหม่ และบันทึกข้อมูลลงในรายการความต้องการของระบบ
50	สรุปปัญหาและความต้องการของผู้ใช้
60	เสนอปัญหาและความต้องการให้กับคณะกรรมการดูแลโครงการ

เทคนิคที่ใช้	ผลลัพธ์ที่ได้
แบบจำลองกระแสข้อมูล (Data Flow Model, DFM)	- แผนภาพกระแสข้อมูลหลักของระบบที่ต้องการ (Context Diagram - Required Environment) - แผนภาพรวมกระแสข้อมูลของระบบงานที่ต้องการ (Overview DFM - Required Environment)
แบบจำลองข้อมูลเชิงตรรก (Logical Data Model, LDM)	- แผนภาพรวมของข้อมูลเชิงตรรกของระบบที่ต้องการ (Overview LDM -Required Environment)
การกำหนดความต้องการ (Requirements Definition)	- รายการความต้องการของระบบ (Requirement Catalogue)
การออกแบบส่วนติดต่อกับผู้ใช้ (Dialogue Design)	- รายการความต้องการของผู้ใช้ (User Catalogue)

รูป 2.4 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 020

ทาสก์ : 10	การสร้างความต้องการขั้นต่ำสุดของระบบใหม่
20	สร้างโครงร่างทางเลือกทางธุรกิจขึ้นมา 3-4 ทางเลือก
30	สร้างโครงร่างทางเลือกทางด้านเทคนิคที่จะใช้ในทางเลือกด้านธุรกิจ
40	รวมโครงร่างทางเลือกด้านธุรกิจกับด้านเทคนิคเพื่อสร้างทางเลือกที่เป็นไปได้ของระบบ
50	ตรวจสอบแต่ละทางเลือกและเพิ่มรายละเอียดของแต่ละทางเลือกอีก
60	สร้างโครงร่างของแผนโครงการสำหรับแต่ละทางเลือกที่เป็นไปได้
70	แสดงทางเลือกให้กับคณะทำงานและเลือกทางเลือกที่เป็นไปได้
80	สร้างแผนสำหรับการดำเนินงาน สำหรับทางเลือกที่ได้เลือก และเป็นทางเลือกที่สามารถทำได้จริง

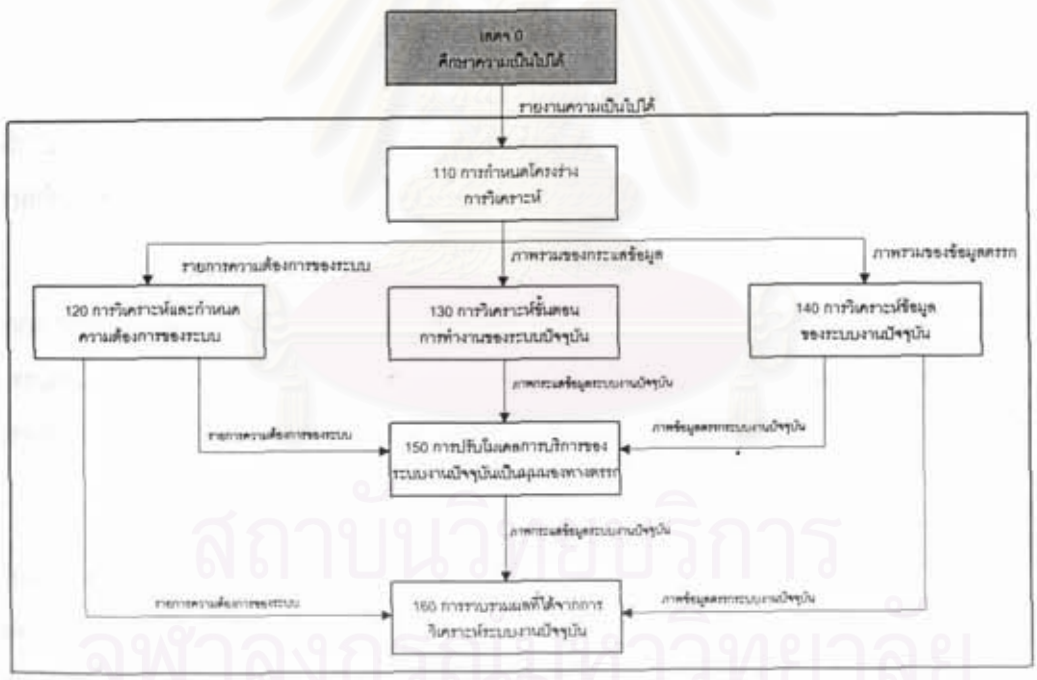
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
ทางเลือกทางด้านธุรกิจ (Business System Option, BSO) ทางเลือกทางด้านเทคนิค (Technical System Option, TSO)	ทางเลือกที่เป็นไปได้

รูป 2.5 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 030

4) เสต็ป 040 รวบรวมรายงานศึกษาความเป็นไปได้ของระบบ เป็นการรวบรวมรายงานหรือผลลัพธ์ที่ได้จากการศึกษาความเป็นไปได้ของระบบ เพื่อเป็นรายงานเสนอให้ผู้บริหาร โดยจะประกอบไปด้วยรายงานต่าง ๆ ดังนี้ รายละเอียดของโครงร่างระบบงานปัจจุบัน รายละเอียดของโครงร่างระบบงานที่ต้องการ รายการปัญหาของระบบ รายการความต้องการของระบบ ทางเลือกที่เป็นไปได้ และแผนสำหรับการดำเนินงาน

2.1.2 เฟสที่ 2 การวิเคราะห์ความต้องการของระบบ (Requirements Analysis) ประกอบด้วย สเตจที่ 1 และ สเตจที่ 2

2.1.2.1 สเตจที่ 1 การศึกษาระบบงานปัจจุบัน สเตจนี้จะอยู่ในขั้นตอนของการวิเคราะห์ (Analysis) ระบบ เริ่มต้นตั้งแต่ศึกษาระบบงานปัจจุบัน และการกำหนดความต้องการของระบบใหม่ โดยในขั้นตอนนี้จะต้องมีการตรวจสอบกันและกันระหว่างแบบจำลองกระแสข้อมูล (DFM) และแบบจำลองข้อมูลเชิงตรรก (LDM) ประกอบไปด้วยเสต็ปย่อย ๆ ดังรูปที่ 2.6 และมีรายละเอียดของแต่ละเสต็ปดังต่อไปนี้



รูป 2.6 แสดงกรอบการทำงานของสเตจ 1

1). เสต็ป 110 กำหนดโครงร่างการวิเคราะห์ จากผลลัพธ์ที่ได้ในขั้นตอนการศึกษาความเป็นไปได้ของระบบจะถูกตรวจสอบถึงขอบเขตของระบบที่ต้องการอีกครั้ง โดยจะทำการทบทวนเอกสารต่าง ๆ และเพิ่มรายละเอียดของความต้องการ หากยังไม่ทำการศึกษาความเป็นไปได้ของระบบ ใน

ขั้นตอนนี้จะเป็นการศึกษาความเป็นไปได้ของระบบ โดยใช้ ทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้จะตรงกับเสต็ป 010 แสดงตามรูป 2.3

2) เสต็ป 120 การวิเคราะห์และกำหนดความต้องการของระบบ ทำการปรับปรุงรายการตามความต้องการของระบบที่ได้รับจาก เสต็ป 110 ให้ตรงกับความต้องการของผู้ใช้ให้มากที่สุด โดยเทคนิคที่ใช้คือ การกำหนดความต้องการ และการออกแบบส่วนติดต่อกับผู้ใช้ โดยการสัมภาษณ์เพิ่มเติม

3) เสต็ป 130 การวิเคราะห์ขั้นตอนการทำงานของระบบปัจจุบัน เสต็ปนี้จะเป็นการแตกลูกหลานขั้นตอนของภาพรวมกระแสข้อมูลของระบบงานปัจจุบันเพื่อให้ทราบข้อมูลขั้นตอนการทำงานของระบบให้ละเอียดขึ้น ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.7

4) เสต็ป 140 การวิเคราะห์ข้อมูลของระบบงานปัจจุบัน เป็นการวิเคราะห์ถึงโครงสร้างข้อมูลทางตรรกโดยจะทำการตรวจสอบ/เพิ่มเติมข้อมูลลงไปในแผนภาพรวม ข้อมูลเชิงตรรกที่ได้จากขั้นตอนการศึกษาความเป็นไปได้ของระบบ โดยโครงสร้างข้อมูลเชิงตรรกในครั้งนี้จะมีเฉพาะข้อมูลระบบงานปัจจุบัน ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.8

5) เสต็ป 150 การปรับแบบจำลองกระแสข้อมูลงานในระบบปัจจุบันเน้นมุมมองทางตรรก เป็นการปรับแบบจำลองกระแสข้อมูลงานในระบบปัจจุบันที่ได้จากเสต็ป 130 ให้เป็นแบบจำลองของการบริการทางตรรก เพื่อจะใช้ในสเตจที่ 3 เพื่อกำหนด ฟังก์ชันของการทำงานให้กับระบบใหม่ ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.9

6) เสต็ป 160 รวมผลที่ได้จากการวิเคราะห์ระบบงานปัจจุบัน เป็นการรวบรวมผลลัพธ์ที่ได้จากการวิเคราะห์ของสเตจนี้ทั้งหมด โดยจะประกอบไปด้วยรายงานต่าง ๆ ดังนี้ แผนภาพกระแสข้อมูลสูงสุด และแผนภาพรวมของข้อมูลทางตรรกของระบบงานปัจจุบัน แบบจำลองกระแสข้อมูลทางตรรก แผนภาพการตรวจสอบไฟล์ข้อมูลกับเอนทิตี้ และรายการความต้องการของระบบและของผู้ใช้

2.1.2.2 สเตจที่ 2 ทางเลือกทางด้านธุรกิจ ในขั้นตอนนี้เป็นการเสนอแนวทางแก้ปัญหาให้กับระบบ โดยพิจารณาถึงผลกระทบและกำไรที่จะได้ ซึ่งจะต้องกำหนดทางเลือกให้กับระบบหลาย ๆ ทางเลือก โดยอาศัยจากข้อมูลที่ได้รับจากสเตจที่ 1 แล้วให้คณะกรรมการดูแลโครงการเป็นคนเลือกแนวทาง ในการแก้ปัญหาที่เหมาะสม ประกอบไปด้วยเสต็ปย่อย ๆ ดังรูปที่ 2.10 และมีรายละเอียดของแต่ละเสต็ปดังต่อไปนี้

1) เสต็ป 210 การกำหนดแนวทางแก้ปัญหาทางธุรกิจ จะเป็นการสร้างทางเลือกหรือแนวทางในการแก้ปัญหาโดยเริ่มสร้างจากความต้องการของระบบขั้นต่ำสุดไปจนถึงขั้นสูงสุด โดยอาจสร้างทางเลือกประมาณ 2-3 ทางเลือก ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.11

2) เสต็ป 220 เลือกแนวทางแก้ปัญหาทางด้านธุรกิจ จะแสดงทางเลือกแต่ละวิธีให้กับคณะกรรมการเพื่อเลือกแนวทางในการแก้ปัญหาที่เหมาะสม

ทาสก์ : 10 วาดแผนภาพการไหลของเอกสาร (Document flow diagram) จากระบบงาน ปัจจุบัน 20 ทำการตรวจสอบและแก้ไขแผนภาพรวมของกระแสข้อมูลของระบบงานปัจจุบัน 30 แตกลูกหลานของภาพรวมของการแสดงข้อมูลของระบบงานปัจจุบันออกเป็นระดับย่อย ๆ 40 สร้างคำอธิบายของเอนทิตี้ภายนอกและคำอธิบายของอินพุต/เอาท์พุต ของระดับล่างสุด ว่าข้อมูลประกอบไปด้วยอะไรบ้าง 50 ปรับปรุง/แก้ไข รายการความต้องการของระบบ ด้วยปัญหาของระบบปัจจุบัน หรือความต้องการของขั้นตอนใหม่ที่ได้รับ	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
แบบจำลองกระแสข้อมูล (Data Flow Model, DFM)	-แผนภาพกระแสข้อมูลหลักของระบบงานปัจจุบัน (Context Diagram - Current Environment) -แผนภาพและคำอธิบายของกระแสข้อมูลในทุกระดับ (Current Physical Data Flow Model)
การกำหนดความต้องการ (Requirements Definition)	- รายการความต้องการของระบบ (Requirement Catalogue)

รูป 2.7 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 130.

ทาสก์ : 10 ตรวจสอบและแก้ไขแผนภาพรวมข้อมูลเชิงตรรกของระบบงานปัจจุบัน 20 กำหนดลักษณะเฉพาะ (Attribute) หลักของแต่ละเอนทิตี้ 30 ทำการตรวจสอบแบบจำลองข้อมูลเชิงตรรกกับแผนภาพ ระดับล่างของขั้นตอนการทำงานว่ามีขั้นตอนที่ทำการสร้างหรือลบเอนทิตี้อยู่ในแบบจำลองกระแสข้อมูลข้อมูลหรือไม่ 40 บันทึกความต้องการข้อมูลหรือปัญหาของระบบงานปัจจุบัน ลงในรายการความต้องการของผู้ใช้	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
แบบจำลองข้อมูลเชิงตรรก (Logical Data Model, LDM)	- แผนภาพรวมของข้อมูลตรรกของระบบงานปัจจุบัน (Logical Data Model -Current Environment)
การกำหนดความต้องการ (Requirement Definition)	- รายการความต้องการของระบบ (Requirement Catalogue)

รูป 2.8 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 140

ทาสก์ : 10 ลบข้อมูลทางกายภาพออกจากแผนภาพกระแสข้อมูล ระดับล่างสุดของระบบงานปัจจุบันที่ได้จากสเต็ป 130 20 สร้างภาพอธิบายไฟล์ของข้อมูลในรูปของเอนทิตี้ 30 สร้างภาพอธิบายขั้นตอนการทำงานระดับล่างสุด ของภาพกระแสข้อมูล 40 ตรวจสอบ/ปรับแต่งแบบจำลองกระแสข้อมูลทางตรรกให้ถูกต้อง 50 ปรับปรุงความต้องการของระบบโดยเพิ่มข้อบังคับทางกายภาพที่ต้องการลงในระบบใหม่	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
แบบจำลองกระแสข้อมูล (Data Flow Model, DFM)	- แบบจำลองกระแสข้อมูลของระบบงานปัจจุบันในมุมมองด้านตรรก (Logical Data Flow Model) - แผนภาพการตรวจสอบไฟล์ข้อมูลกับเอนทิตี้ (Logical Data Store/Entity Cross Reference)
การกำหนดความต้องการ (Requirement Definition)	- รายการความต้องการของระบบ (Requirement Catalogue)
แบบจำลองข้อมูลเชิงตรรก (Logical Data Model, LDM)	- แผนภาพของข้อมูลตรรกของระบบงานปัจจุบัน (Current Environment LDM)

รูป 2.9 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 150

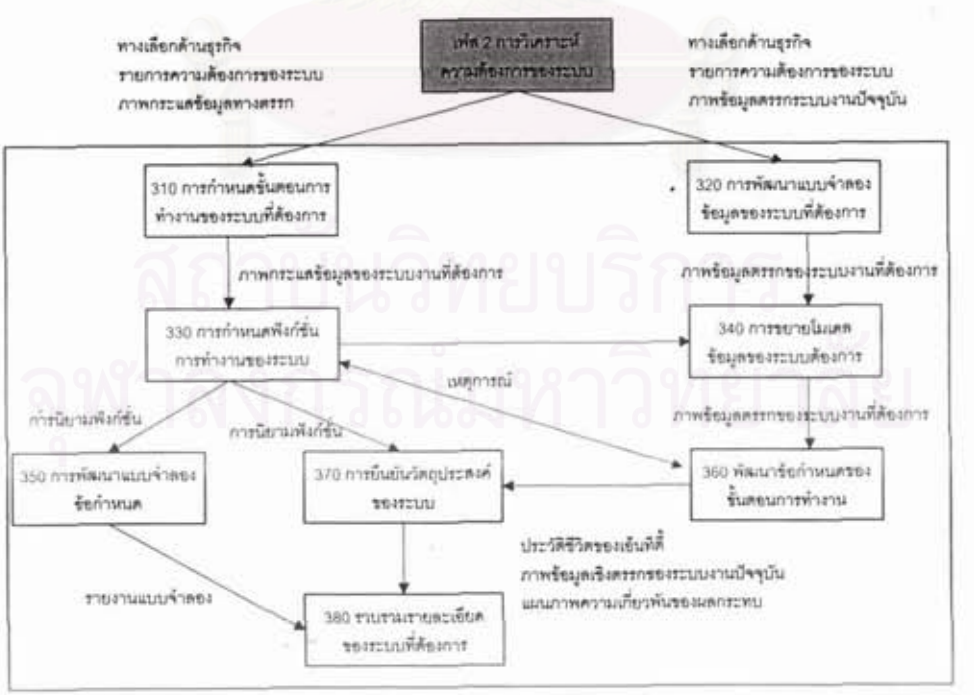
รูป 2.10 แสดงกรอบการทำงานของสเตจ 2

ทาสก์ : 10 สร้างความต้องการขั้นต่ำสุดของระบบใหม่ที่ต้องการในแต่ละทางเลือก 20 สร้างฟังก์ชันพื้นฐานที่ต้องการทั้งหมด 30 สรุปแนวทางสั้น ๆ ของแต่ละทางเลือก 40 เพิ่มข้อมูลรายละเอียดของแนวทางลงไปในทางเลือก เช่น ราคา, ผลกระทบ, แผนของงาน 50 เปรียบเทียบราคา/กำไร และผลกระทบที่ได้รับจากแนวทางแก้ไขแต่ละวิธี	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
ทางเลือกทางด้านธุรกิจ โดยอาจใช้ แบบจำลองกระแสข้อมูล และแบบจำลองข้อมูลตรรก เป็นตัวกำหนดทางเลือก	- ทางเลือก/แนวทางในการแก้ปัญหา (Business System Options)

รูป 2.11 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 210

2.1.3 เฟสที่ 3 ข้อกำหนดเฉพาะของระบบที่ต้องการ (Requirements Specification) ประกอบด้วย สเตจที่ 3 เพียงสเตจเดียว

สเตจที่ 3 การศึกษาความต้องการของระบบ สเตจนี้ถือเป็นหัวใจของการวิเคราะห์และออกแบบ SSADM ซึ่งจะเป็นการจัดทำรายละเอียดของความต้องการของระบบเพื่อให้มีความถูกต้องว่าระบบที่ผู้ใช้ต้องการนั้นเป็นอย่างไร โดยอาศัยข้อมูลจากรายการความต้องการที่ได้รับมาจากสเตจที่ 1 และสเตจที่ 2 ประกอบไปด้วยเสต็ปย่อย ๆ ดังรูปที่ 2.12 และมีรายละเอียดของแต่ละเสต็ปดังต่อไปนี้



รูป 2.12 แสดงกรอบการทำงานสเตจ 3

1) เสต็ป 310 การกำหนดขั้นตอนการทำงานของระบบที่ต้องการ เพื่อเป็นการปรับปรุงและเพิ่มเติมรูปแบบจำลองกระแสข้อมูลตามแนวทางแก้ปัญหาที่ได้เลือก และจะทำการแปลงรายการความต้องการของระบบให้อยู่ในแผนภาพกระแสข้อมูลของระบบที่ต้องการ ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.13

2) เสต็ป 320 การพัฒนาแบบจำลองข้อมูลของระบบที่ต้องการ เพื่อปรับปรุงและเพิ่มเติมแบบจำลองข้อมูลเชิงตรรกตามรายการความต้องการ ให้มีความถูกต้องยิ่งขึ้น ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.14

3) เสต็ป 330 การกำหนดฟังก์ชันการทำงานของระบบ เป็นการแปลงขั้นตอนการทำงานที่อยู่ในรูปของแผนภาพกระแสข้อมูล ให้อยู่ในรูปของโครงสร้างการเชื่อมโยงระหว่างมนุษย์กับคอมพิวเตอร์ของระบบที่ต้องการ เพื่อให้สามารถใช้ในคอมพิวเตอร์ได้ ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.15

ทาสก์ : 10	ปรับปรุง/แก้ไขรายการความต้องการของระบบ ตามที่เลือกจากแนวทางแก้ปัญหาทางธุรกิจที่เลือกจากเสต็ป 220
20	ปรับปรุง/แก้ไขแผนภาพรวมของกระแสข้อมูล โดยการเพิ่มขั้นตอนการทำงานที่ต้องการในระบบใหม่ตามแนวทางแก้ปัญหาทางธุรกิจที่เลือก
30	ปรับปรุงแก้ไขแผนภาพของกระแสข้อมูลระดับต่าง ๆ โดยทำการตรวจสอบกับความต้องการของระบบที่มีการปรับปรุงแล้ว
40	ปรับปรุง/แก้ไขรายละเอียดของขั้นตอนการทำงานระดับล่าง รายละเอียดอินพุต / เอาท์พุต และรายละเอียดเอนทิตี้ภายนอก ตามแผนภาพของกระแสข้อมูลที่เปลี่ยนแปลง
50	ตรวจสอบว่าแหล่งเก็บข้อมูลตรงกับเอนทิตี้ที่อยู่ในแผนภาพรวมของข้อมูลเชิงตรรกของระบบที่ต้องการหรือไม่
60	กำหนดบทบาทหน้าที่ของผู้ใช้ (User Roles) และตรวจสอบว่าตรงกับเอนทิตี้ภายนอกที่ในแผนภาพของกระแสข้อมูล

เทคนิคที่ใช้	ผลลัพธ์ที่ได้
รูปแบบจำลองกระแสข้อมูล (Data Flow Model, DFM)	- แผนภาพกระแสข้อมูลของระบบที่ต้องการ (Required System Data Flow Model) - แผนภาพการตรวจสอบไฟล์ข้อมูลกับเอนทิตี้ (Logical Data Store /Entity Cross Reference)
การออกแบบส่วนติดต่อกับผู้ใช้ (Dialogue Design)	- รายการแสดงบทบาทหน้าที่ของผู้ใช้ (User Role)
การกำหนดความต้องการ (Requirements Definition)	- รายการความต้องการของระบบ (Requirements Catalogue)

รูป 2.13 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 310

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ทาสก์ : 10 ปรับปรุง/แก้ไขแผนภาพข้อมูลดรรก โดยการเพิ่มข้อมูลที่ต้องการในระบบใหม่ ตามแนวทางการแก้ปัญหาทางธุรกิจที่เลือก นอกจากนี้จะต้องจัดทำเอกสารหรือรายละเอียดของเอนทิตี้(Entity Description) และรายละเอียดของคุณลักษณะเฉพาะ (Attribute / Data Item Description) โดยทำการตรวจสอบกับรายการความต้องการ (Requirements Catalogue Entry) 20 ตรวจสอบว่าแผนภาพข้อมูลดรรกที่ได้นั้น มีการกล่าวถึงในแผนภาพกระแสข้อมูลระดับล่างของระบบที่ต้องการหรือไม่ 30 ปรับปรุงแก้ไขแผนภาพข้อมูลดรรกตามความต้องการที่เป็นฟังก์ชั่น (Functional) และไม่เป็นฟังก์ชั่น (Non-Functional) จากรายการความต้องการ	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
รูปแบบจำลองข้อมูลเชิงดรรก (Logical Data Model, LDM)	- แผนภาพข้อมูลดรรกของระบบที่ต้องการ (Required System Logical Data Model) - รายละเอียดของเอนทิตี้(Entity Description) - รายละเอียดของลักษณะเฉพาะ (Attribute/Data Item Description)
การกำหนดความต้องการ (Requirements Definition)	- รายการความต้องการของระบบ (Requirements Catalogue)

รูป 2.14 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 320

ทาสก์ : 10 เลือกฟังก์ชันปรับปรุง (Update Function) คือฟังก์ชันที่มีการแก้ไขข้อมูล จากแผนภาพกระแสข้อมูลโดยการแนะนำของผู้ใช้ โดยตรวจสอบว่าแต่ละขั้นตอนการทำงานระดับโปรเซสพื้นฐาน (Elementary Process) จะต้องมีอย่างน้อย 1 ฟังก์ชัน นอกจากนี้จะต้องเลือกส่วนประกอบของฟังก์ชัน คือ ว่าเหตุการณ์(event) ใดที่เกิดขึ้น และมีการสอบถาม (enquiry) ในฟังก์ชันเกิดขึ้นหรือไม่ 20 เลือกฟังก์ชันสอบถาม (enquiry function) จากรายการความต้องการ (Requirements Catalogue) และจากขั้นตอนการทำงานที่เป็นงานสอบถามข้อมูลจากแผนภาพกระแสข้อมูล (Data Flow Model) 30 สร้างแผนภาพโครงสร้างอินพุต / เอาท์พุต (I/O Structure Diagram) ของแต่ละฟังก์ชัน และรายละเอียดของโครงสร้างของอินพุต / เอาท์พุต (I/O Structure Description) 40 สร้างตารางเมตริกแสดงหน้าที่ /บทบาทของผู้ใช้ (User Role / Fonction Matrix) และระบุว่าส่วนใดของฟังก์ชันที่ซับซ้อน 50 ระบุถึงระดับการให้บริการของแต่ละฟังก์ชัน	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
การนิยามฟังก์ชัน (Function Definition)	- ตารางแสดงนิยามของฟังก์ชัน (Function Definition) - แผนภาพและรายละเอียดของโครงสร้างอินพุต / เอาท์พุต (I/O Structure Diagram and I/O Structure Description)
การออกแบบส่วนติดต่อกับผู้ใช้ (Dialogue Design)	- ตารางเมตริกแสดงหน้าที่ /บทบาทของผู้ใช้ (User Role / Function Matrix)
การนิยามความต้องการ (Requirement Definition)	- รายการความต้องการ (Requirements Catalogue)

รูป 2.15 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 330

4) เสต็ป 340 การขยายแบบจำลองข้อมูลของระบบที่ต้องการ ในเสต็ปนี้จะเป็นการใช้เทคนิควิเคราะห์ข้อมูลเชิงสัมพันธ์ (Relational Data Analysis, RDA) เข้าช่วยในการวิเคราะห์ข้อมูล วิธีนี้จะทำให้ได้ผลลัพธ์ของข้อมูลเพิ่ม ผู้ใช้จะต้องทำการตรวจสอบ และนำข้อมูลที่เพิ่มเข้าไปใส่ในแผนภาพข้อมูลตรรก ประกอบด้วยทาสก์ ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.16

5) เสต็ป 350 พัฒนาแบบจำลองข้อกำหนด (Specification Prototypes) ก่อนที่จะมีการพัฒนาระบบทั้งหมด จะต้องสร้างแบบจำลองขึ้นมาเพื่อตรวจสอบความต้องการของผู้ใช้ โดยจะนำส่วนที่มีความซับซ้อนขึ้นมาพัฒนาก่อน และผลที่ได้จะนำกลับไปพิจารณาในการกำหนด ฟังก์ชันและรายการความต้องการ ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.17

6) เสต็ป 360 พัฒนาข้อกำหนดของลำดับขั้นตอนการทำงาน (Processing Specification) โดยนำขั้นตอนการทำงานที่ได้จากแผนภาพกระแสข้อมูล และนิยามฟังก์ชัน และข้อมูลที่ต้องการที่ได้จากแผนภาพข้อมูลเชิงตรรก และจากการวิเคราะห์ข้อมูลเชิงสัมพันธ์ มาศึกษาถึงผลกระทบของเวลา หรือเหตุการณ์ที่เกิดขึ้นเมื่อเวลาเปลี่ยนไป เสต็ป 360 นี้ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.18

ทาสก์ : 10 เลือกแผนผังโครงสร้างอินพุต/เอาท์พุต ที่จะใช้ทำรูปแบบบรรทัดฐาน (Normalization) 20 ทำรูปแบบบรรทัดฐานตั้งแต่ระดับที่ 1 ถึงระดับที่ 3 30 ใช้ตารางที่ได้จากการทำรูปแบบบรรทัดฐานนำมาสร้างแผนภาพข้อมูลตรรกขึ้นมาใหม่ 40 เปรียบเทียบแผนภาพข้อมูลตรรก ที่สร้างจากตารางที่ผ่านการรูปแบบบรรทัดฐาน กับแผนภาพข้อมูลตรรกที่ระบบต้องการในส่วนที่แตกต่างกัน	
เทคนิคที่ใช้	**ผลลัพธ์ที่ได้**
วิเคราะห์ข้อมูลเชิงสัมพันธ์ (Relational Data Analysis, RDA)	- แผนภาพข้อมูลเชิงตรรกของระบบที่ต้องการโดยสร้างจากตารางที่ได้จากการทำรูปแบบบรรทัดฐาน (Required System , LDM)

รูป 2.16 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 340

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ทาสก์ : 10 เลือกรายการที่ต้องการทำแบบจำลอง 20 จัดทำผังงานโครงสร้างเมนู (Menu Structure Chart) จำลองขึ้น 30 สร้างทางเข้าถึงของแบบจำลองจากเมนูลงไปถึงส่วนที่ติดต่อกับผู้ใช้ทั้งหมด ทำทาสก์ที่ 40-70 ซ้ำในแต่ละฟังก์ชันที่นำมาสร้างแบบจำลอง 40 สร้างแบบจำลองของทางเดิน 50 ทำแบบจำลองทั้งหมด 60 แสดงแบบจำลองที่ได้ให้กับกลุ่มผู้ใช้ 70 ทบทวนแบบจำลอง 80 ปรับปรุงข้อกำหนดความต้องการที่ได้รับจากการทำแบบจำลอง เพื่อให้ได้ผลลัพธ์ที่ผู้ใช้ต้องการ และจัดทำรายงานของแบบจำลอง	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
แบบจำลองข้อกำหนด (Specification Prototypes)	- รายงานของแบบจำลอง (Prototyping Report)
การออกแบบส่วนติดต่อกับผู้ใช้ (Dialogue Design)	- ผังงานโครงสร้างเมนู (Menu Structures) - ผังงานโครงสร้างคำสั่ง (Command Structure)
การนิยามความต้องการ (Requirements Definition)	- รายการความต้องการ (Requirements Catalogue)

รูป 2.17 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 350

ทาสก์ : 10 เลือกเหตุการณ์ที่มีผลกระทบในแต่ละเอนทิตี้ โดยการใช้ตารางแสดงความสัมพันธ์ระหว่างเหตุการณ์/เอนทิตี้ (Event/Entity Matrix) 20 สร้างประวัติชีวิตของแต่ละเอนทิตี้ จากข้างล่างขึ้นด้านบนของแผนภาพข้อมูลดรรก และสร้างจากด้านบนลงล่างของแบบจำลองข้อมูลเชิงตรรก โดยการเพิ่มเหตุการณ์ผิดปกติหรือข้อยกเว้นลงในประวัติชีวิตของเอนทิตี้ และเพิ่มขั้นตอนการทำงานลงในประวัติชีวิตของเอนทิตี้ 30 สร้างแผนภาพความเกี่ยวพันของผลกระทบ (Effect Correspondence Diagram) สำหรับแต่ละเหตุการณ์ 40 ปรับปรุง/แก้ไขรายการความต้องการและแบบจำลองข้อมูลดรรก 50 สร้างทางเข้าถึงการสอบถามสำหรับแต่ละฟังก์ชันสอบถาม 60 เพิ่มอัตราส่วนจำนวนของเหตุการณ์ที่เกิดขึ้นในแต่ละเอนทิตี้และแต่ละความสัมพันธ์	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
แบบจำลองเอนทิตี้เหตุการณ์ (Entity-Event Modelling)	- ประวัติชีวิตของเอนทิตี้ (Entity Life Histories, ELH) - แผนภาพความเกี่ยวพันของผลกระทบ (Effect Correspondence Diagrams, ECD)
การนิยามความต้องการ (Requirements Definition)	- รายการความต้องการ (Requirements Catalogue)

รูป 2.18 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 360

7) เสต็ป 370 การยืนยันวัตถุประสงค์ของระบบ ในระหว่างการวิเคราะห์ความต้องการของระบบโดยมีการ เพิ่ม/ปรับปรุงรายการความต้องการของผู้ใช้อยู่ตลอดเวลา ดังนั้นเสต็ปนี้เป็นการตรวจ

สอบความต้องการที่เป็นฟังก์ชันหรือไม่เป็นฟังก์ชัน เพื่อให้แน่ใจว่าตรงกับเอนทิตี้ในรายการความต้องการ เสต็ป 370 นี้ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.19

8) เสต็ป 380 รวบรวมรายละเอียดที่ต้องการของระบบ เป็นการรวบรวมรายละเอียดที่ต้องการที่ถูกจัดทำขึ้นในสเตจนี้ เพื่อจัดทำข้อกำหนดของระบบที่ต้องการ จะประกอบด้วยรายงาน แผนภาพข้อมูลเชิงตรรกของระบบที่ต้องการ ตารางแสดงนิยามของฟังก์ชัน แผนภาพและรายละเอียดโครงสร้างอินพุต/เอาท์พุต ประวัติชีวิตของเอนทิตี้ แผนภาพความเกี่ยวพันของผลกระทบ เส้นทางการเข้าถึงการสอบถาม รายการความต้องการของระบบ และตารางแสดงบทบาทและหน้าที่การทำงานของผู้ใช้

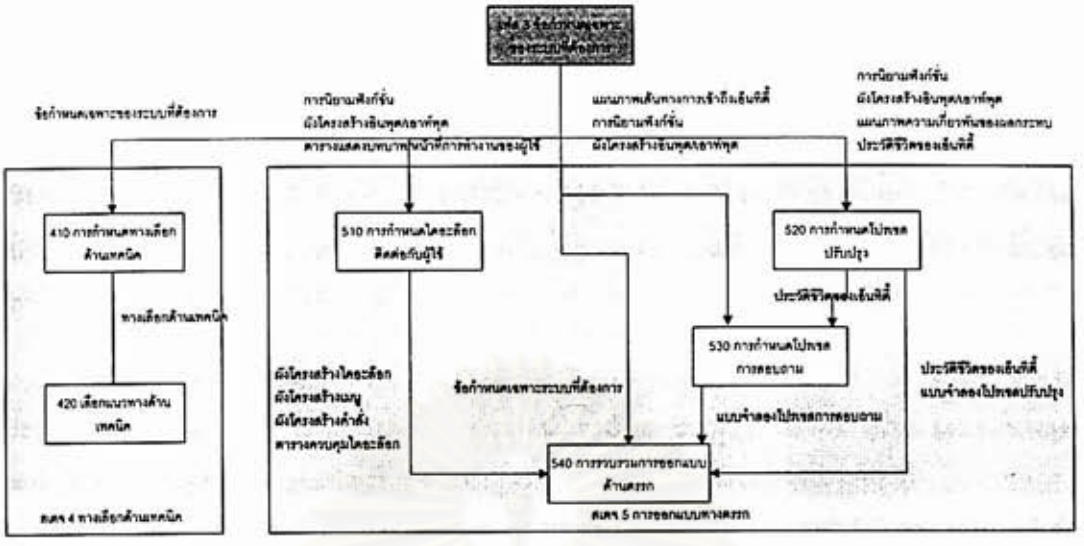
ทาสก์ : 10	ตรวจสอบว่าทุกความต้องการที่เป็นฟังก์ชันและไม่เป็นฟังก์ชันได้กำหนดแล้ว โดยการตรวจสอบกับรายการความต้องการ
20	กำหนดความต้องการใหม่/ที่มีอยู่เดิม ที่ไม่เป็นฟังก์ชันลงในรายการความต้องการ, รายการนิยามฟังก์ชัน หรือแบบจำลองข้อมูลตรรกของระบบที่ต้องการ
30	ตรวจสอบว่าแต่ละนิยามของฟังก์ชันมีครบถ้วน
40	ตรวจสอบว่าแบบจำลองข้อมูลของระบบที่ต้องการมีการกำหนดครบแล้ว

เทคนิคที่ใช้	ผลลัพธ์ที่ได้
นิยามฟังก์ชัน (Function Definition)	- ตารางแสดงนิยาม ของฟังก์ชัน (Function Definition)
แบบจำลองข้อมูลเชิงตรรก (Logical Data Model, LDM)	- แผนภาพข้อมูลเชิงตรรกของระบบที่ต้องการ (Required System LDM)
การนิยามความต้องการ (Requirements Definition)	- รายการความต้องการ (Requirements Catalogue)

รูป 2.19 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 370

2.1.4 เฟสที่ 4 ข้อกำหนดเฉพาะของระบบเชิงตรรก (Logical System Specification) ประกอบด้วยสเตจ 4 และสเตจ 5 โดยสามารถทำขนานกันได้ทั้งสองสเตจ

2.1.4.1 สเตจที่ 4 ทางเลือกด้านเทคนิค เป็นการตอบคำถามว่าระบบใหม่ที่ต้องการพัฒนานั้นอยู่ในสภาพแวดล้อมทางเทคนิคอย่างไร รวมไปถึงเทคนิค/วิธีการพัฒนาระบบโดยจะกำหนดทางเลือกให้ผู้บริหารโครงการเป็นคนเลือก ประกอบไปด้วยเสต็ปย่อย ๆ ดังรูปที่ 2.20 และมีรายละเอียดของแต่ละเสต็ปดังต่อไปนี้



รูป 2.20 แสดงกรอบการทำงานสเตจ 4 และสเตจ 5

1) เสต็ป 410 การกำหนดทางเลือกด้านเทคนิค เป็นการเสนอแนวทางด้านเทคนิคให้ผู้ใช้ทั้งทางด้านฮาร์ดแวร์, ซอฟต์แวร์ และเป้าหมายของการพัฒนา ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.21

2) เสต็ป 420 การเลือกทางเลือกด้านเทคนิค เป็นการเลือกแนวทางด้านเทคนิคที่มีการเสนอว่าต้องการอย่างไร โดยดูจากผลกระทบ , ต้นทุน , และผลที่จะได้รับ

ทาสก์ : 10 กำหนดข้อบังคับด้านเทคนิค 20 สร้างโครงร่างของทางเลือกด้านเทคนิค 30 แสดงข้อมูลให้ผู้ใช้ทราบและลดทางเลือกให้เหลือ 2-3 ข้อ 40 เพิ่มรายละเอียดของทางเลือกแต่ละวิธี โดยการเพิ่มรายละเอียดสภาพแวดล้อมด้านเทคนิค และของระบบ 50 ตรวจสอบว่าแต่ละทางเลือก ให้บริการตามความต้องการครบหรือไม่ 60 กำหนดรายละเอียดแต่ละทางเลือก โดยการเพิ่มการวิเคราะห์ผลกระทบ, แผนร่างการพัฒนา และการวิเคราะห์ต้นทุน/กำไร	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
ทางเลือกด้านเทคนิค (Technical System Options, TSO)	- ทางเลือกด้านเทคนิค

รูป 2.21 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 410

2.1.4.2 สเตจที่ 5 การออกแบบทางตรรก (Logical Design) ในระหว่างการออกแบบในวงจรการพัฒนาแบบจะต้องนำความต้องการที่ได้จากการวิเคราะห์มาเพื่อผลิตลำดับขั้นตอนการทำงานทางด้านตรรก เพื่อนำเสนอระบบใหม่ โดยผลลัพธ์ที่ได้ในขั้นตอนนี้จะทราบว่าระบบที่ได้จะทำงานอย่างไร ในสถานการณ์ต่าง ๆ แต่จะไม่แสดงถึงขั้นตอนการทำงาน หรือข้อกำหนดเฉพาะของโปรแกรม (Program Specifications) และการออกแบบในขั้นกำหนดรายละเอียดของระบบที่จะได้ โดยไม่ขึ้นกับฮาร์ดแวร์ต่าง ๆ ซึ่งประกอบไปด้วยเสต็ปย่อย ๆ ตามรูปที่ 2.20 และมีรายละเอียดของแต่ละเสต็ปดังต่อไปนี้

1) เสต็ป 510 การกำหนดส่วนติดต่อกับผู้ใช้ เป็นส่วนที่ใช้แทนการโต้ตอบระหว่างผู้ใช้กับระบบ โดยจะเป็นการกำหนดและออกแบบส่วนประกอบที่จะปรากฏในหน้าจอ รายงาน ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.22

2) เสต็ป 520 การกำหนดโปรเซสปรับปรุง (Update Process) เพื่อสร้างภาพของผลกระทบของเหตุการณ์ในรูปแบบจำลองข้อมูลตรรกของระบบ เพื่อให้ทราบว่าระบบจะควบคุมผลกระทบที่เกิดขึ้นของแต่ละเหตุการณ์อย่างไร ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.23

3) เสต็ป 530 การกำหนดโปรเซสสอบถาม (Enquiry Processes) ใช้การสร้างผังงานโครงสร้างโปรเซส การสอบถามซึ่งจะพัฒนาโดยการเชื่อมแผนภาพเส้นทางการเข้าถึงเอนทิตี้ และแผนผังโครงสร้างอินพุต/เอาท์พุต ประกอบด้วยทาสก์ต่าง ๆ เทคนิคและผลลัพธ์ที่ได้ดังรูป 2.24

<table>
<tr><td colspan="2">ทาสก์ : 10 แปลงโครงสร้างอินพุต/เอาท์พุต ให้อยู่ในรูปโครงสร้างไดอะล็อก และทำการเลือก/จัดกลุ่มของสมาชิกภายในโครงสร้างไดอะล็อกที่มีหน้าที่เดียวกันเข้าไว้ด้วยกัน (Logical Groupings of Dialogue Element, LGDE)
20 ทำการหาเส้นทางที่เป็นไปได้โดยผ่านไดอะล็อกในตารางควบคุมไดอะล็อก
30 สร้างแผนภาพโครงสร้างเมนูสำหรับผู้ใช้แต่ละกลุ่ม (User Role) และเอกสารของเส้นทางที่เป็นไปได้จากโครงสร้างของแผนภาพโครงสร้างคำสั่ง (Command Structures)
40 พิจารณาว่าควรมีการสร้างไดอะล็อกสำหรับความช่วยเหลือที่ใดบ้าง</td></tr>
<tr><td>เทคนิคที่ใช้</td><td>ผลลัพธ์ที่ได้</td></tr>
<tr><td>การออกแบบส่วนติดต่อกับผู้ใช้ (define User Dialogues)</td><td>-โครงสร้างส่วนติดต่อกับผู้ใช้ (Dialogue Structure)
-โครงสร้างเมนู (Menu Structure)
- โครงสร้างคำสั่ง (Command Structure) ตารางควบคุมผังงานส่วนติดต่อกับผู้ใช้
-ส่วนให้ความช่วยเหลือกับผู้ใช้ (Dialogue Level Help)</td></tr>
</table>

รูป 2.22 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 510

ทาสก์ : 10 เพิ่มตัวชี้สถานะลงไปยังแผนภาพประวัติชีวิตเอนทิตี้ เพื่อใช้ตามร่องรอยของเอนทิตี้ ทำทาสก์ 20 ถึง 50 สำหรับแต่ละเหตุการณ์ (Event) 20 สร้างแผนภาพโครงสร้างการปรับปรุง สำหรับแต่ละเหตุการณ์ โดยการใช้แผนภาพความเกี่ยวพันของผลกระทบ และผังโครงสร้างเอาท์พุต 30 แจกแจงการทำงานของแต่ละเหตุการณ์ 40 กำหนดการทำงาน (Operations) ให้กับโครงสร้างของโปรเซสสำหรับแต่ละเหตุการณ์ และกำหนดเงื่อนไขทางเลือก (Selection) และการทำซ้ำ(Iterations) ให้กับผัง งานโครงสร้างของโปรเซส 50 กำหนดข้อความ (Message) ที่จะปรากฏเมื่อมีความผิดพลาดเกิดขึ้น	
เทคนิคที่ใช้	**ผลลัพธ์ที่ได้**
แบบจำลองเหตุการณ์เอนทิตี้ (Entity Event Modelling)	- แผนภาพประวัติชีวิตเอนทิตี้ (ระบุตัวชี้สถานะ) (ELHs)
แบบจำลองข้อมูลเชิงตรรก (LDM)	- แผนภาพข้อมูลเชิงตรรก (ระบุตัวชี้สถานะ)
การออกแบบโปรเซส ระดับฐานข้อมูลเชิงตรรก (Logical Database Process Design, LDPD)	- แบบจำลองโปรเซสปรับปรุง

รูป 2.23 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 520

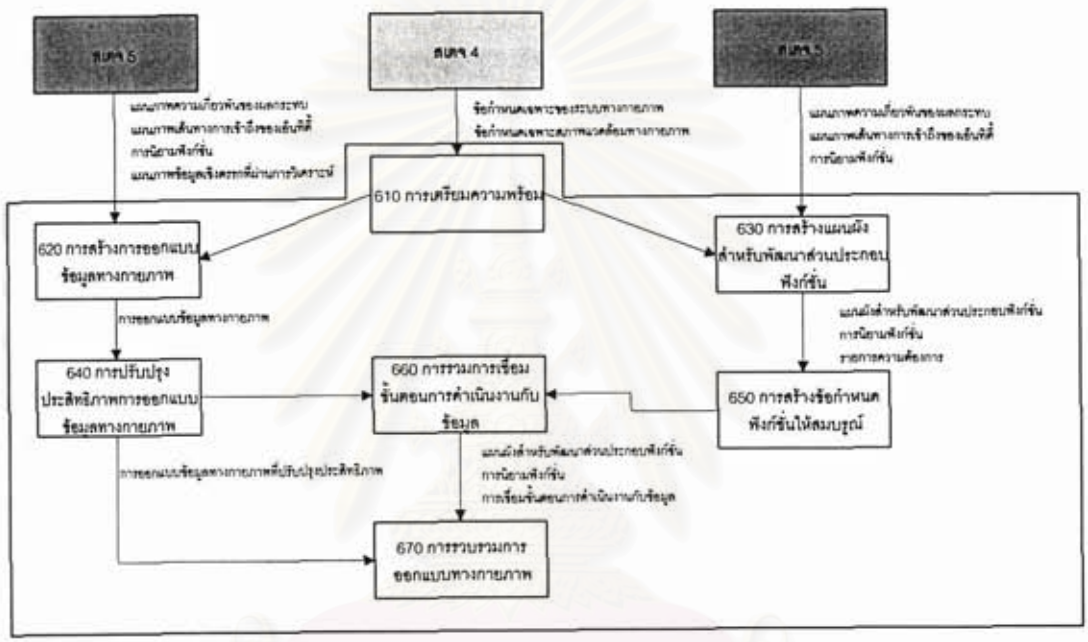
ทำทาสก์ 10-30 สำหรับแต่ละการสอบถาม ทาสก์ : 10 แปลงแผนภาพเส้นทางการเข้าถึงเอนทิตี้ (Enquiry Access Path, EAP) ให้อยู่ในรูปแผนผังโครงสร้างโปรเซสสำหรับอินพุตของการสอบถาม 20 แปลงแผนผังโครงสร้างเอาท์พุต ของโครงสร้างอินพุต/เอาท์พุตให้อยู่ในรูปแผนผังโครงสร้างโปรเซส สำหรับอินพุต/เอาท์พุตของการสอบถาม 30 นำแผนผังโครงสร้างอินพุตและโครงสร้างเอาท์พุตทั้ง 2 มาเชื่อมกัน	
เทคนิคที่ใช้	**ผลลัพธ์ที่ได้**
การออกแบบโปรเซส ระดับฐานข้อมูลเชิงตรรก(LDPD)	- แบบจำลองโปรเซสการสอบถาม

รูป 2.24 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 530

4) เสต็ป 540 การรวบรวมการออกแบบทางด้านตรรก เป็นการรวบรวมรายละเอียดแผนผังที่ถูกจัดทำขึ้นในขั้นตอนการออกแบบของสเตจนี้ทั้งหมด ประกอบไปด้วยรายงานผังแสดงส่วนติดต่อกับผู้ใช้ ตารางควบคุมเข้างานผังโครงสร้างเมนู ผังงานโครงสร้างคำสั่ง แบบจำลองโปรเซสปรับปรุง และแบบจำลองโปรเซสการสอบถาม

2.1.5 เฟสที่ 5 การออกแบบทางกายภาพ (Physical Design) ประกอบด้วย สเตจที่ 6 เพียงสเตจเดียว

สเตจที่ 6 การออกแบบทางกายภาพ เป็นการแปลงข้อกำหนดเฉพาะของระบบทางด้านตรรก ให้อยู่ในรูปแบบของสภาพแวดล้อมระบบที่เราต้องการ โดยการศึกษาประสิทธิภาพการทำงาน, ปริมาณเนื้อที่ของการใช้งาน และเวลาที่ใช้ในการพัฒนางาน เพื่อให้ได้การพัฒนาระบบที่มีประสิทธิภาพ ประกอบไปด้วยเสต็ปย่อย ๆ ดังรูปที่ 2.25 และมีรายละเอียดของแต่ละเสต็ปดังต่อไปนี้



รูป 2.25 แสดงกรอบการทำงานสเตจ 6

1) เสต็ป 610 การเตรียมความพร้อมสำหรับการออกแบบทางกายภาพ เป็นการศึกษาถึงเครื่องมือที่จะใช้ในการพัฒนาโปรแกรมเพื่อปรับสถานการณ์ในข้อมูลต่าง ๆ กับทีมงานประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.26

2) เสต็ป 620 การสร้างการออกแบบข้อมูลทางกายภาพ เป็นการเปลี่ยนแบบจำลองข้อมูลตรรกของระบบที่ต้องการ ให้อยู่ในรูปทางด้านกายภาพว่ามีการจัดการหรือเก็บในรูปแบบใด โดยในขั้นตอนนี้เป็นการทำการออกแบบข้อมูลเบื้องต้น ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.27

3) เสต็ป 630 การสร้างแผนผังสำหรับการพัฒนาส่วนประกอบฟังก์ชัน (Function Component Implementation Map, FCIM) เป็นการสร้างข้อกำหนดของแต่ละฟังก์ชัน ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ ดังรูป 2.28

4) เสต็ป 640 การปรับปรุงประสิทธิภาพการออกแบบข้อมูลทางกายภาพ เป็นการปรับปรุงการออกแบบข้อมูลเบื้องต้นโดยเน้นให้มีประสิทธิภาพ เช่น การกำหนดที่เก็บข้อมูล, เวลา และวัตถุประสงค์ของระบบ ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.29

5) เสต็ป 650 การสร้างข้อกำหนดของฟังก์ชันให้สมบูรณ์ ข้อกำหนดของโปรแกรม (Program Specification) จะถูกนำมาเขียนให้อยู่ในรูปแบบโปรแกรม โดยจะใช้ข้อกำหนดเฉพาะการออกแบบกระบวนการทำงานทางกายภาพ (Physical Process Specification) เพื่อสร้างแผนผังพัฒนาส่วนประกอบของฟังก์ชัน รายการความต้องการ และนิยามฟังก์ชันให้สมบูรณ์

6) เสต็ป 660 การรวมเชื่อมขั้นตอนการดำเนินงานกับข้อมูล (Process Data Interface, PDI) เป็นการสร้างรอยต่อระหว่างการออกแบบฐานข้อมูลในรูปแบบทางตรรกมาเป็นการออกแบบฐานข้อมูลทางกายภาพ ประกอบด้วยทาสก์ต่าง ๆ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ดังรูป 2.30

7) เสต็ป 670 การรวบรวมการออกแบบทางกายภาพ เป็นการรวบรวมรายละเอียดที่ถูกจัดทำขึ้นในสเตจนี้ เพื่อจัดทำรายงานของระบบที่ต้องการ ประกอบไปด้วยเอกสาร ผังสำหรับการพัฒนาส่วนประกอบฟังก์ชัน การออกแบบข้อมูลทางกายภาพ ข้อมูลตรรกของระบบที่ต้องการ ข้อกำหนดขั้นตอนการดำเนินงานกับข้อมูล ประมาณเนื้อที่และเวลาที่ใช้

ทาสก์ : 10 เลือกสภาพแวดล้อมทางกายภาพที่ต้องการ เช่น ระบบประมวลผลที่จะใช้, ที่เก็บข้อมูล และระบบบริหารฐานข้อมูลที่ต้องการ 20 พิจารณาเวลาและเนื้อที่ที่จะใช้ของระบบบริหารฐานข้อมูล โดยเปรียบเทียบกับหลาย ๆ ระบบบริหารฐานข้อมูล 30 กำหนดมาตรฐานของสภาพแวดล้อมทางกายภาพของการประมวลผลและสิ่งที่อำนวยความสะดวกของข้อมูล 40 ติดตั้งกฎ/ข้อกำหนดการออกแบบข้อมูล 50 กำหนดมาตรฐานการตั้งชื่อของโปรแกรม, ชื่อตารางข้อมูล 60 จัดทำเป้าหมายการออกแบบทางกายภาพและข้อกำหนดของระบบ 70 เตรียมความพร้อมให้กับผู้ใช้โดยการอบรมและให้ลองปฏิบัติ 80 เสนอแนวทาง/เป้าหมายให้กับผู้บริหารโครงการ	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
การออกแบบข้อมูลทางกายภาพ โดยกำหนดกระบวนการทางกายภาพ (Physical Process Specification)	- มาตรฐานการพัฒนาโปรแกรมแอปลิเคชัน (Application Development Standards) - เป้าหมายการออกแบบทางกายภาพ (Physical Design Strategy)

รูป 2.26 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 610

ทาสก์ : 10 เปลี่ยนแบบจำลองข้อมูลตรรกของระบบที่ต้องการ ให้อยู่ในรูปแบบจำลองข้อมูลทางกายภาพ โดยการลบสมาชิกที่ไม่จำเป็นสำหรับการออกแบบทางกายภาพ 20 เลือกจุดทางเข้าที่ต้องการของเอนทิตี้และคีย์ที่ใช้ 30 เลือกเอนทิตี้ที่เป็นหลักในการเข้าถึงสำหรับลำดับขั้นตอนที่เป็น 40 จัดกลุ่มของเอนทิตี้ที่เป็นลูก ๆ ของเอนทิตี้หลัก 50 ในบางครั้งเอนทิตี้ที่ไม่ใช่รูทอาจถูกจัดกลุ่มมากกว่า 1 กลุ่ม จะต้องเลือกให้อยู่ในกลุ่มเดียว 60 ติดตั้งขนาดของบล็อก (Block Size) 70 แปงกลุ่มทางกายภาพให้พอดีกับบล็อก 80 นำข้อกำหนดการออกแบบข้อมูลไปใช้กับส่วนอื่น	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
การออกแบบข้อมูลทางกายภาพ (Physical Data Design)	- การออกแบบข้อมูลเบื้องต้น (1st-cut) - การประเมินเนื้อที่ที่ต้องการ(Space Estimation)

รูป 2.27 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 620

ทาสก์ : 10 ตรวจสอบทุก ๆ ฟังก์ชันเพื่อที่จะเลือกและลบบางโมดูล, จอภาพที่ซ้ำ 20 ตรวจสอบทุก ๆ ฟังก์ชันเพื่อเลือกและขยายการใช้โมดูลหรือจอภาพที่ใช้ร่วมกัน ทำทาสก์ 30-80 สำหรับแต่ละฟังก์ชัน 30 กำหนดข้อมูลที่ถูกต้องเมื่อรับข้อมูลเข้า 40 กำหนดข้อมูลที่ผิดพลาดเมื่อรับข้อมูลเข้า 50 กำหนดการควบคุมและจัดการข้อผิดพลาดที่เกิดขึ้น 60 เลือกรูปแบบทางกายภาพสำหรับส่วนที่ติดต่อกับระบบทั้งหมด 70 ออกแบบส่วนประกอบทางกายภาพของแต่ละไดอะล็อก รวมถึงรูปแบบรายงาน, จอภาพ, เมนู 80 พัฒนาส่วนประกอบฟังก์ชันในส่วนที่ไม่ได้เป็นโปรซีเดอร์โดยการใช้เครื่องมือที่ช่วยในการอำนวยความสะดวก เช่น การสร้างหน้าจอหรือรายงาน	
เทคนิคที่ใช้	ผลลัพธ์ที่ได้
ข้อกำหนดโปรเซส ทางกายภาพ (Physical Data Design)	- เขียนผังสำหรับการพัฒนาส่วนประกอบฟังก์ชัน (Function Component Implementation Map ,FCIM) - นิยามฟังก์ชัน (Function Definitions)

รูป 2.28 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 630

ทาสก์ : 10 ปรับปรุงประสิทธิภาพของการออกแบบข้อมูลทางกายภาพในด้านเนื้อที่การเก็บข้อมูล 20 ปรับปรุงประสิทธิภาพของการออกแบบข้อมูลทางกายภาพในด้านเวลาของการค้นหา, การเข้าถึงข้อมูล รวมไปถึงโครงสร้างข้อมูล	
เทคนิคที่ใช้	**ผลลัพธ์ที่ได้**
การออกแบบข้อมูลทางกายภาพ (Physical Data Design)	- การออกแบบข้อมูลทางกายภาพ (Physical Data Design) - การนิยามฟังก์ชัน (Function Definition) - รายการความต้องการ (Requirement Catalogue) - ประเมินเนื้อที่ที่ต้องการ (Space Estimations) - ประเมินเวลาที่ใช้ในการเข้าถึงข้อมูล/ การประมวลผล (Timing Estimations)

รูป 2.29 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 640

ทาสก์ : 10 เลือกสิ่งที่ไม่เหมือนกันระหว่างมุมมองทางตรรกของข้อมูลที่อยู่ในข้อกำหนดของฟังก์ชัน 20 เลือกคีย์ของทุก ๆ รายชื่อข้อมูลที่ไม่เหมือนกัน 30 กำหนดวิธีการที่จะใช้เชื่อมจากมุมมองหนึ่งไปมุมมองหนึ่ง เพื่อแก้ปัญหาข้อผิดพลาดต่าง ๆ 40 ตรวจสอบว่าไม่มีสมาชิกของตัวเชื่อมขั้นตอนการดำเนินงานกับข้อมูลซ้ำกัน 50 จัดทำเอกสารโปรเซสการเชื่อมขั้นตอนการดำเนินงานกับข้อมูลในผังสำหรับการพัฒนาส่วนประกอบฟังก์ชัน 60 ปรับปรุงรายการความต้องการที่มีการเปลี่ยนแปลงเพื่อแสดงที่เก็บข้อมูลที่ต้องการ	
เทคนิคที่ใช้	**ผลลัพธ์ที่ได้**
ข้อกำหนดโปรเซส ทางกายภาพ (Physical Process Specification)	- ส่วนติดต่อการดำเนินงานกับข้อมูล (Process Data Interface, PDI) - แผนผังสำหรับการพัฒนาส่วนประกอบฟังก์ชัน (FCIM) - รายการความต้องการ (Requirement Catalogue) - นิยามฟังก์ชัน (Function Definitions)

รูป 2.30 แสดงทาสก์ เทคนิคที่ใช้ และผลลัพธ์ที่ได้ของเสต็ป 660

2.2 กลุ่มของเครื่องมือ/เทคนิคมาตรฐานที่ใช้ในการวิเคราะห์และออกแบบ

กลุ่มของเครื่องมือ/เทคนิคที่ใช้ในการวิเคราะห์และออกแบบเชิงโครงสร้าง SSADM มีหลายวิธี และถูกใช้ในแต่ละเสต็ป ซึ่งจะได้พูดถึงแต่ละเทคนิคดังต่อไปนี้

2.2.1 แบบจำลองกระแสข้อมูล (Data Flow Model, DFM) เป็นเทคนิคที่ถูกใช้ในหลาย ๆ ทาสก์ และใช้กันมากในการวิเคราะห์โดยทั่ว ๆ ไป โดยเฉพาะวิธีการวิเคราะห์ในเชิงโครงสร้าง เนื่องจากเป็นเทคนิคที่เข้าใจง่าย ประกอบด้วย 2 ส่วน คือ แผนภาพกระแสข้อมูล (Data Flow Diagram, DFD) และคำอธิบายรายละเอียดของแผนภาพ โดยจะเป็นการแสดงถึงการส่งผ่านข้อมูลรอบ ๆ ระบบ ส่วนประกอบของแผนภาพกระแสข้อมูล มีดังนี้

1) เอนทิตี้ภายนอก (External Entity) คือบุคคล, องค์กรต่าง ๆ หรือระบบคอมพิวเตอร์อื่น ๆ ที่ส่งข้อมูลเข้าหรือออกจากระบบ โดยไม่สนใจการทำงานของสิ่งที่อยู่ภายนอก สัญลักษณ์ที่ใช้จะแสดงตามรูป 2.31

2) โปรเซส (Process) คือขั้นตอนที่จะต้องทำ โดยจะเปลี่ยนข้อมูลเข้าเป็นผลลัพธ์ซึ่งชื่อโปรเซสจะเป็นตัวบอกว่าโปรเซสนี้มีหน้าที่ทำอะไร ซึ่งคำที่ใช้จะใช้คำกริยา สัญลักษณ์ที่ใช้แสดงตามรูป 2.31

3) แหล่งเก็บข้อมูล (Data Storage) ข้อมูลจะถูกเก็บในไฟล์หรือในแฟ้มเอกสาร ต่าง ๆ และถูกใช้เมื่อต้องการ สัญลักษณ์ที่ใช้ตามรูป 2.31 แบ่งประเภทของแหล่งเก็บข้อมูลได้ 4 แบบ ดังนี้

D แทนแหล่งเก็บข้อมูลที่ได้จากการประมวลด้วยคอมพิวเตอร์

M แทนแหล่งเก็บข้อมูลที่เป็นตู้เก็บเอกสาร, แฟ้มเอกสาร

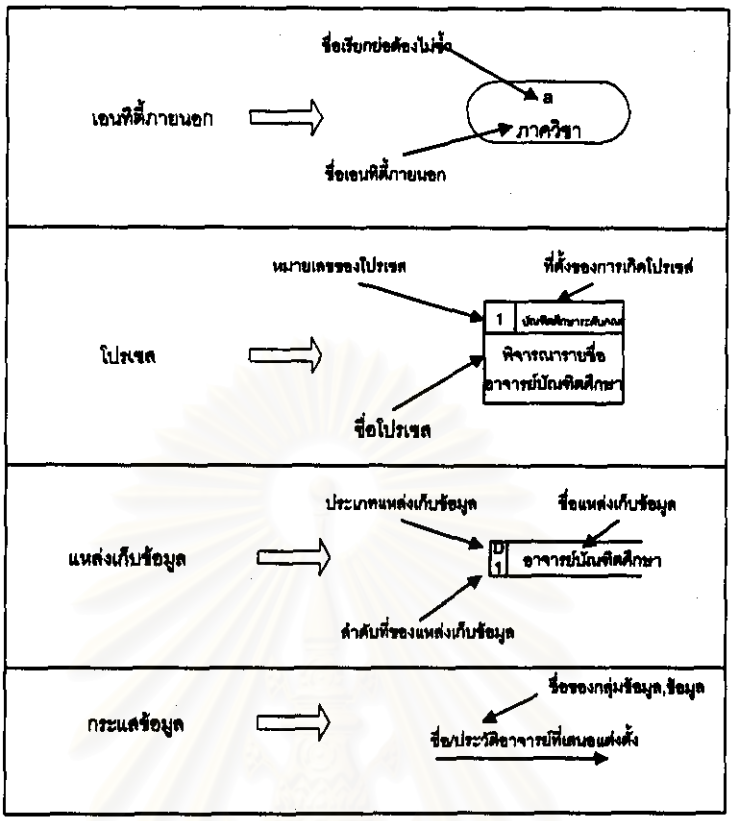
T(M) แทนแหล่งเก็บข้อมูลชั่วคราวแต่ข้อมูลนี้เป็นข้อมูลที่อยู่ในรูปของเอกสาร

T แทนแหล่งเก็บข้อมูลชั่วคราวที่ได้จากคอมพิวเตอร์ เช่น ไฟล์ที่ได้จากการจัดลำดับ

4) กระแสข้อมูล (Data Flow) ข้อมูลจะมีการไหลระหว่างโปรเซสต่าง ๆ และอาจจะมีการเคลื่อนที่มาจากสิ่งที่อยู่นอกระบบ มีสัญลักษณ์ที่ใช้แสดงตามรูป 2.31

การเชื่อมต่อกับกระแสข้อมูลนี้ทำได้เฉพาะ ระหว่าง 2 โปรเซส, ระหว่างแหล่งเก็บข้อมูลลงโปรเซส และระหว่างโปรเซสกับเอนทิตี้ภายนอกเท่านั้น สำหรับขั้นตอนการพัฒนาแผนภาพกระแสข้อมูลแสดงตามรูป 2.32

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูป 2.31 แสดงสัญลักษณ์ที่ใช้ของแผนภาพกระแสข้อมูล

ขั้นตอนการพัฒนาแผนภาพกระแสข้อมูล

1) สร้างตารางและหาเอนทิตี้ภายนอกและหาว่าเอนทิตี้แต่ละตัวสร้างข้อมูลและรับข้อมูลอะไรบ้าง
2) แปลงตารางที่ได้ในข้อ 1 ให้เป็นแผนภาพกระแสข้อมูลหลัก (Context Diagram)
3) สร้างตารางและหาเอกสารที่เกิดขึ้นในระบบ รวมทั้งหาว่าเอนทิตี้ใดที่สร้างและรับเอกสารเหล่านี้บ้าง
4) นำตารางที่ได้จากข้อ 3 มาเขียนแผนภาพการส่งผ่านเอกสาร (Document Flow Diagram)
5) แปลง/เปลี่ยนแผนภาพการส่งผ่านเอกสารให้เป็นแผนภาพกระแสข้อมูล (Overview Data Flow Diagram)
6) ทำการแตกระดับของแผนภาพรวมกระแสข้อมูลลงสู่ระดับล่าง ซึ่งบางโปรเซส อาจจะมีขั้นตอนการทำงานที่ซับซ้อน การแตกระดับจะทำให้ทราบถึงขั้นตอนการทำงานได้ดียิ่งขึ้น การแตกจนถึงระดับล่างสุดจะเรียกโปรเซสนั้นว่า โปรเซส พื้นฐาน (Elementary Process) นอกจากนี้จัดทำคำอธิบายรายละเอียดของโปรเซสพื้นฐาน (Elementary Process Description, EPD)

รูป 2.32 แสดงขั้นตอนการพัฒนาแผนภาพกระแสข้อมูล

2.2.2 แบบจำลองข้อมูลเชิงตรรก (Logical Data Modelling, LDM) เป็นเทคนิคที่ใช้ในการวิเคราะห์มุมมองทางด้านข้อมูลว่าข้อมูลมีโครงสร้างอย่างไร และมีความสัมพันธ์กันอย่างไร ประกอบไปด้วย 2 ส่วน คือ แผนภาพโครงสร้างข้อมูลเชิงตรรก (Logical Data Structure, LDS) และคำอธิบายรายละเอียดของแผนภาพ (Textual Descriptions) โดยจะถูกใช้ทั้งการวิเคราะห์ข้อมูลของระบบปัจจุบันและระบบที่ต้องการ สำหรับขั้นตอนการพัฒนาแบบจำลองข้อมูลเชิงตรรก ตามรูป 2.33 แบบจำลองข้อมูลเชิงตรรกประกอบด้วย

1) เอนทิตี้ (Entities) กลุ่มของข้อมูลจะถูกแทนด้วยรูปสี่เหลี่ยมผืนผ้า โดยข้อมูลที่เหมือนกันจะเรียกว่าเป็นลักษณะเฉพาะ (Attribute)

2) ความสัมพันธ์ (Relationship) แต่ละเอนทิตี้จะต้องมีความสัมพันธ์กันโดยจะถูกเชื่อมโยงกันด้วยเส้น และต้องมีชื่อของความสัมพันธ์

3) ระดับความสัมพันธ์ของสองเอนทิตี้ แบ่งได้ 3 ประเภท ดังนี้

-ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One-to-One Relationships) เป็นความสัมพันธ์ของข้อมูลของเอนทิตี้หนึ่งว่ามีความสัมพันธ์กับข้อมูลอย่างมากหนึ่งข้อมูลกับอีกเอนทิตี้หนึ่งในลักษณะที่เป็นหนึ่งต่อหนึ่ง สัญลักษณ์ที่ใช้แทนตามรูป 2.34

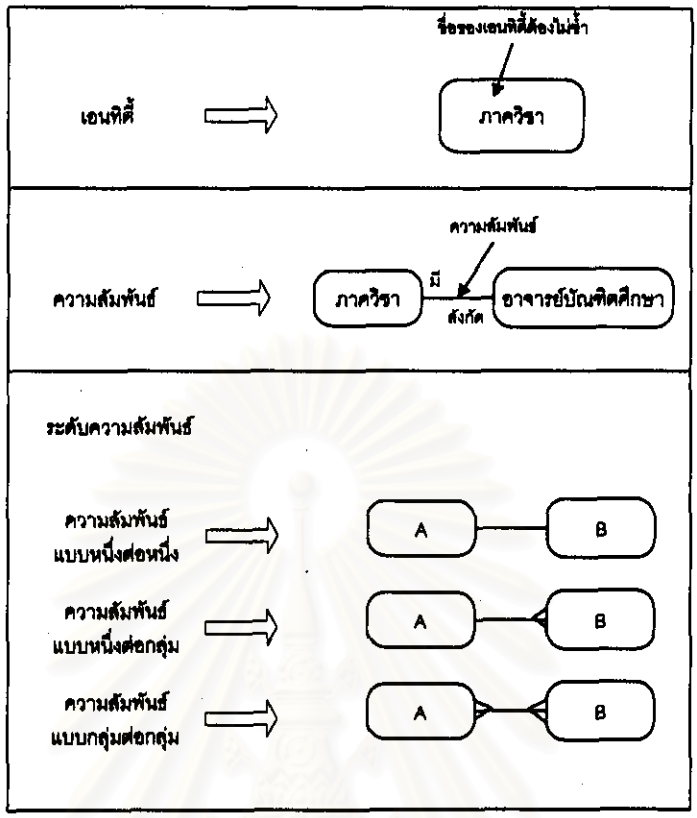
-ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One-to-Many Relationships) เป็นความสัมพันธ์ของข้อมูลของเอนทิตี้หนึ่งว่ามีความสัมพันธ์กับข้อมูลหลายข้อมูลกับอีกเอนทิตี้หนึ่งในลักษณะที่เป็นหนึ่งต่อหนึ่ง สัญลักษณ์ที่ใช้แทนตามรูป 2.34

-ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many-to-Many Relationships) เป็นความสัมพันธ์ของสองเอนทิตี้ในลักษณะแบบกลุ่มต่อกลุ่ม สัญลักษณ์ที่ใช้แทนตามรูป 2.34

ขั้นตอนการพัฒนาแผนภาพโครงสร้างข้อมูลเชิงตรรก

1) กำหนดเอนทิตี้ทั้งหมดของระบบ
2) สร้างตารางเมตริกเอนทิตี้เพื่อหาความสัมพันธ์ระหว่างเอนทิตี้
3) กำหนดความสัมพันธ์ระหว่างเอนทิตี้ จะทำให้ได้แผนภาพโครงสร้างข้อมูลเชิงตรรก

รูป 2.33 แสดงขั้นตอนการพัฒนาแบบจำลองข้อมูลเชิงตรรก



รูป 2.34 แสดงสัญลักษณ์ที่ใช้ของแผนภาพข้อมูลเชิงตรรก

2.2.3 การออกแบบส่วนติดต่อกับผู้ใช้ (Dialogue Design) เป็นเทคนิคที่ใช้กำหนดรูปแบบการติดต่อระหว่างระบบกับผู้ใช้ โดยการกำหนดแบบและหน้าที่ให้กับผู้ใช้ ประกอบด้วย 2 ขั้นตอนสำคัญ คือ การกำหนดส่วนติดต่อกับผู้ใช้ (Dialogue Identification) ตามรูป 2.35 และการออกแบบส่วนติดต่อกับผู้ใช้ (Dialogue Design) ตามรูป 2.36

ขั้นตอนการกำหนดส่วนติดต่อกับผู้ใช้

1) รายการแสดงหน้าที่ผู้ใช้ (User Catalogue) โดยการสร้างตารางผู้ใช้ทั้งหมดในระบบ และบันทึกหน้าที่ที่ผู้ใช้ได้รับแต่ละคน
2) รายการแสดงบทบาทหน้าที่ของผู้ใช้ (User Role) ในระบบใหม่ โดยการสร้างตารางผู้ใช้และหน้าที่ในระบบใหม่
3) ตารางระหว่างผู้ใช้กับฟังก์ชันการทำงาน (User Role / Function Matrix) ภายหลังจากที่มีการสร้างฟังก์ชันการทำงานในระบบจะต้องมีการสร้างตารางเพื่อกำหนดการเข้าใช้ของผู้ใช้ในแต่ละฟังก์ชัน
4) กำหนดไดอะล็อกที่วิกฤต (Critical Dialogue) จากตารางที่ได้จากข้อ 3 เลือกส่วนที่วิกฤต เช่น มีการใช้บ่อย ๆ, มีเอนทิตี้ที่ใช้ในฟังก์ชันนั้นเป็นจำนวนมาก โดยการวางกลมไปยังหน้าที่ดังกล่าว

รูป 2.35 แสดงขั้นตอนการกำหนดส่วนติดต่อกับผู้ใช้

I17792652X

ขั้นตอนการออกแบบส่วนติดต่อกับผู้ใช้

1) สร้างรายละเอียดของสมาชิกไดอะล็อก (Dialogue Element Description) โดยใช้แผนภาพโครงสร้างไดอะล็อก (Dialogue Structure) และคำนิยามฟังก์ชัน (Function Definition)
2) จัดกลุ่ม/รวมกลุ่มของสมาชิกของไดอะล็อกที่เหมือนกันไว้ด้วยกัน โดยใช้แผนภาพโครงสร้างไดอะล็อก (Dialogue Structure)
3) สร้างตารางควบคุมไดอะล็อกเพื่อหาว่าผู้ใช้นิยมใช้การเข้าถึงใดมากกว่ากัน

รูป 2.36 แสดงขั้นตอนการออกแบบส่วนติดต่อกับผู้ใช้

2.2.4 การนิยามความต้องการของระบบ (Requirements Definition) เป็นขั้นตอนการหาความต้องการของผู้ใช้ในระบบ ซึ่งจะถูกใช้ในหลายเสต็ปของการทำงาน ซึ่งมีขั้นตอนการนิยามความต้องการของระบบ ตามรูป 2.37

ขั้นตอนการนิยามความต้องการของระบบ

1) สอบถามความต้องการของผู้ใช้ในระบบใหม่ที่ต้องการ ทั้งในแง่ของฟังก์ชันทำงานที่ต้องการ (Functional Requirements) และไม่ใช่ฟังก์ชันการทำงาน (Non-Functional Requirement) เช่น เวลาของการตอบสนองของฟังก์ชัน, ความปลอดภัยและการเข้าถึง, ราคาของระบบ
2) จัดทำรายการความต้องการ (Requirement Catalogue) ในรูปของตารางและคำอธิบายความต้องการ
3) จัดทำรายการความต้องการ (Requirements Catalogue) ที่จะถูกทำในระบบใหม่

รูป 2.37 ขั้นตอนการนิยามความต้องการของระบบ

2.2.5. การวิเคราะห์ข้อมูลเชิงสัมพันธ์ (Relational Data Analysis, RDA) เป็นเทคนิคที่อยู่ใช้ในเสต็ป 340 การขยายแบบจำลองข้อมูลที่ต้องการเป็นแบบจำลองเชิงสัมพันธ์ (Relational Model) มีการจัดเก็บข้อมูลในลักษณะตาราง ประกอบด้วยตารางหลายตารางที่มีความสัมพันธ์กัน โดยตารางเป็นตาราง 2 มิติ มีการจัดเก็บข้อมูลแต่ละช่องของตาราง ข้อมูลในแนวนอนหมายถึงระเบียนข้อมูล (Row) ซึ่งจะต้องไม่ซ้ำกัน นอกจากนี้ลำดับของแถวจะต้องไม่ซ้ำกัน ความมาก่อนหลังไม่สำคัญ โดยมีข้อมูลในแนวสดมภ์เป็นเขตข้อมูล เรียกว่าลักษณะเฉพาะ (Attribute) โดยแต่ละลักษณะเฉพาะต้องมีชื่อไม่ซ้ำกัน และแต่ละสดมภ์ของลักษณะเฉพาะต้องมีข้อมูลไม่ซ้ำกัน สดมภ์ไหนมาก่อนมาหลังไม่สำคัญ นอกจากนี้ต้องมีการกำหนดคีย์ที่เกี่ยวข้อง เช่น คีย์หลัก (Primary Key) นอกจากนั้นต้องมีการทำการจัดรูปแบบบรรทัดฐาน (Normalize) ของข้อมูล เพื่อให้ได้แบบจำลองข้อมูลที่มีความเหมาะสมถูกต้อง ไม่ขัดแย้งกัน ขั้นตอนการพัฒนาแบบจำลองเชิงสัมพันธ์ มีดังนี้

1) เขียนชื่อเอนทิตี้ (Entities)

2) เขียนลักษณะเฉพาะ (Attribute) ทั้งหมดที่อยู่ภายในเอนทิตี้

3) หาลักษณะเฉพาะที่เป็นคีย์หลัก แล้วขีดเส้นใต้ที่ลักษณะเฉพาะดังกล่าว

4) ทำการรูปแบบบรรทัดฐาน

5) นำตารางที่ได้จากการทำรูปแบบบรรทัดฐานมาเขียนแบบจำลองข้อมูลเชิงตรรก

การจัดรูปแบบบรรทัดฐาน (Normalization)

การจัดรูปแบบบรรทัดฐานเป็นขบวนการพัฒนา ตรวจสอบ และกลั่นกรองแบบจำลองข้อมูลในรูปแบบของตารางความสัมพันธ์ หรือบางครั้งมักเรียกโดยย่อว่าความสัมพันธ์ ให้เป็นไปตามนิยามของการรูปแบบบรรทัดฐาน โดยมีจุดประสงค์เพื่อ

- ลดปัญหาซ้ำซ้อนของข้อมูล
- ลดที่ว่างที่ต้องใช้ในการจัดเก็บข้อมูล
- ลดความผิดพลาด ความไม่ตรงกันของข้อมูลในฐานข้อมูล
- ลดปัญหาการเกิดรูปแบบบรรทัดฐานของการลบ และการแก้ไขข้อมูล
- เพิ่มความคงทนให้แก่โครงสร้างของฐานข้อมูล

การทำรูปแบบบรรทัดฐานแบ่งได้เป็นหลายระดับ แต่ในการวิเคราะห์และออกแบบ SSADM จะใช้ 3 ระดับ ซึ่งมีรายละเอียดแต่ละระดับ ตามรูป 2.38

การจัดทำรูปแบบบรรทัดฐาน

1) รูปแบบบรรทัดฐานระดับที่ 1 (First Normal Form, 1NF)
ตาราง (Table) ที่อยู่ในนอร์มัลฟอร์มค่าลักษณะเฉพาะจะต้องไม่เป็นกลุ่มหรืออยู่ซ้ำซ้อนกัน จะต้องทำการแยกตารางโดยตารางที่แยกจะต้องมีคีย์หลักจากตารางเดิมรวมกับลักษณะเฉพาะอื่นเพื่อให้แถวของข้อมูลนั้นไม่ซ้ำ

2) รูปแบบบรรทัดฐานระดับที่ 2 (Second Normal Form, 2NF)
ตาราง (Table) นี้จะต้องอยู่ในกฎนอร์มัลฟอร์มระดับที่ 1 และทุกลักษณะเฉพาะจะต้องขึ้นอยู่กับคีย์หลัก

3) รูปแบบบรรทัดฐานระดับที่ 3 (Third Normal Form, 3NF)
ตาราง (Table) นี้จะต้องอยู่ในกฎนอร์มัลฟอร์มระดับที่ 2 ลักษณะเฉพาะทุกลักษณะเฉพาะที่ไม่ได้เป็นคีย์จะต้องไม่ขึ้นกับลักษณะเฉพาะที่เป็นคีย์คู่แข่ง (Candidate Key)

รูป 2.38 แสดงการจัดทำรูปแบบบรรทัดฐาน

2.2.6 การนิยามฟังก์ชัน (Function Definition) เป็นเทคนิคที่ช่วยในการสร้างส่วนที่ใช้ติดต่อระหว่างเครื่องคอมพิวเตอร์กับมนุษย์, ลำดับขั้นตอนของการดำเนินงาน โดยการสร้างข้อกำหนด

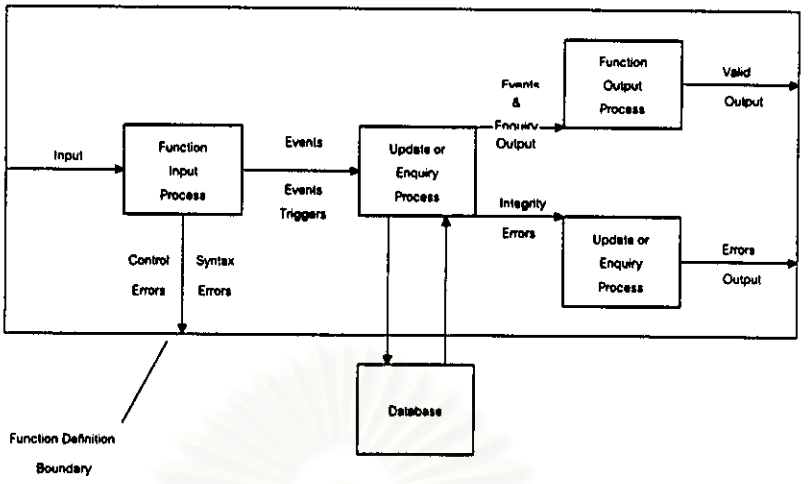
ของการประมวลผลทำได้โดยการนิยามฟังก์ชัน ฟังก์ชัน เป็นกลุ่มของขั้นตอนการทำงานในเวลาเดียวกัน นอกจากนี้ยังสร้างขึ้นมาเพื่อจัดการกับผลของเหตุการณ์ในระบบ ทั้งนี้ฟังก์ชันเป็นหน่วยงานพื้นฐานของการทำงาน

ประเภทของฟังก์ชัน แบ่งได้เป็น 3 กลุ่ม ดังนี้ กลุ่มแรก คือฟังก์ชันสอบถามหรือปรับปรุง (Enquiry or Update) ฟังก์ชันปรับปรุงใช้ในเมื่อมีการเปลี่ยนแปลงข้อมูลของระบบ กลุ่มที่สอง ออนไลน์หรือออฟไลน์ (On-line or Off-line) และกลุ่มสุดท้ายดูว่าแหล่งใดเป็นผู้สร้างข้อมูล เช่น ผู้ใช้หรือระบบเป็นแหล่งกำเนิดข้อมูล (User or System Initiated) แบบจำลองฟังก์ชันทั่วไป (Universal Function Model) แสดงตามรูป 2.39 ประกอบไปด้วยส่วนต่าง ๆ ดังนี้

1) อินพุต (Input) ข้อมูลที่นำเข้ามาใช้ในฟังก์ชัน
2) เอาท์พุตที่ถูกต้อง (Valid Output) ผลลัพธ์ที่ได้จากฟังก์ชัน
3) ขอบเขตของการนิยามฟังก์ชัน (Function Definition Boundary)
4) ผลลัพธ์ที่ได้จากเหตุการณ์และจากการสอบถาม (Event and Enquiry Output)
5) โปรเซสการปรับปรุง (Update Process) เป็นโปรเซสที่ทำให้ข้อมูลมีการเปลี่ยนแปลง
6) โปรเซสสอบถาม (Enquiry Process) เป็นโปรเซสที่มีการสอบถามข้อมูล
7) ข้อผิดพลาดของข้อมูลที่เกิดจากการทำงานของโปรแกรม (Integrity Errors)
8) ความผิดพลาดทางโครงสร้างของภาษา (Syntax and Control Errors)
9) โปรเซสอินพุต/เอาท์พุต (I/O process)
10) เอาท์พุตที่ผิดพลาด (Error Output) หรือผลลัพธ์ที่ผิดพลาดจากฟังก์ชัน

ขั้นตอนการสร้างนิยามฟังก์ชัน รายละเอียดของฟังก์ชัน (Function Description) รายละเอียดอินพุต/เอาท์พุต (I/O Description) แผนภาพโครงสร้างอินพุต/เอาท์พุต (I/O Structure Diagram) และรายละเอียดของโครงสร้างอินพุต/เอาท์พุต จะแสดงตามรูป 2.40

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูป 2.39 แสดงส่วนประกอบของฟังก์ชัน แบบจำลองฟังก์ชันสากล

ขั้นตอนการสร้างนิยามฟังก์ชัน

1) เลือกฟังก์ชันปรับปรุงจากแบบจำลองกระแสข้อมูลของระบบที่ต้องการที่ระดับล่างสุด (Elementary Process)
2) เลือกฟังก์ชันสอบถามจากเอกสารในรายการความต้องการ
3) ทำการตรวจสอบฟังก์ชันกับผู้ใช้
4) จัดทำเอกสารของฟังก์ชันเกี่ยวกับรายละเอียดของฟังก์ชัน
5) จัดสร้างรายละเอียดส่วนติดต่อกับผู้ใช้ โดยสร้างรายละเอียดอินพุต/เอาท์พุต (I/O Description) โดยสร้างจากกระแสข้อมูล (Data Flows) ที่อยู่ภายในแบบจำลองกระแสข้อมูลระดับล่างว่าประกอบรายการข้อมูลอะไรบ้าง
6) สร้างแผนภาพโครงสร้างส่วนติดต่อกับผู้ใช้ที่เรียกว่าโครงสร้างอินพุต/เอาท์พุต (I/O Structure Diagram) และรายละเอียดของโครงสร้างอินพุต/เอาท์พุต (I/O Structure Description)

ขั้นตอนการสร้างแผนภาพโครงสร้างอินพุต/เอาท์พุต

1) เขียนสมาชิกของโครงสร้างอินพุต/เอาท์พุต (I/O Structure Elements) เป็นรูปสี่เหลี่ยมที่มุมมนที่อยู่ด้านล่างสุดของแผนภาพ ใช้แทนกลุ่มของรายการข้อมูล โดยต้องระบุว่าเป็นกลุ่มของรายการที่เป็นอินพุตหรือเอาท์พุต
2) เขียนชื่อโครงสร้าง (Structure Boxes) เป็นกล่องสี่เหลี่ยมที่ไม่ได้อยู่ล่างสุดของแผนภาพ โดยกล่องสี่เหลี่ยมบนสุดจะแทนชื่อของแผนภาพโครงสร้างอินพุต/เอาท์พุต และส่วนอื่นจะถูกใช้แทนวิธีที่รายการข้อมูลถูกสร้างภายในแผนภาพ
3) ลำดับของการแสดงข้อมูลจะอ่านจากแผนภาพล่างสุดจากซ้ายไปขวา
4) บางครั้งการติดต่อกับผู้ใช้ อาจจะต้องถูกเลือกให้ทำอย่างใดอย่างหนึ่ง จะเพิ่มสัญลักษณ์ รูปวงกลม ที่มุมบนด้านขวา ของรูปสี่เหลี่ยมมุมมน
5) บางส่วนอาจมีการวนลูป เช่น ส่วนที่เป็นฟอร์มย่อยของรายการหลัก จะเพิ่มสัญลักษณ์ รูป * ที่มุมบนด้านขวา ของรูปสี่เหลี่ยมมุมมน
6) เขียนรายละเอียดของโครงสร้างอินพุต/เอาท์พุต ของแต่ละสมาชิกโครงสร้าง

รูป 2.40 สดงขั้นตอนการสร้างนิยามฟังก์ชันและแผนภาพโครงสร้างของอินพุต/เอาท์พุต

2.2.7 การสร้างแบบจำลองข้อกำหนดเฉพาะ (Specification Prototyping) ก่อนที่จะมีการเขียนข้อกำหนดเฉพาะของโครงการทั้งหมดในเสต็ป 350 จะเลือกบางส่วนเพื่อสร้างแบบจำลองให้ผู้ใช้ได้ดูก่อน เพื่อตรวจสอบถึงความต้องการของผู้ใช้ที่เราได้รับมาครบถ้วนหรือไม่ หากโครงการนั้นมีความเสี่ยงและมีราคาสูง โดยมีขั้นตอนการพัฒนาแบบจำลองข้อกำหนดเฉพาะตามรูป 2.41

ขั้นตอนการพัฒนาแบบจำลองข้อกำหนดเฉพาะ

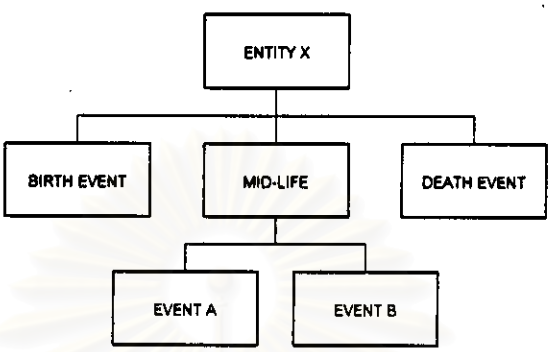
1) กำหนดขอบเขตของแบบจำลอง โดยเลือกจากตารางบทบาทหน้าที่ของผู้ใช้/ ฟังก์ชัน โดยเลือกเฉพาะส่วนที่วิกฤตมาทำแบบจำลองก่อน เช่น เลือกส่วนที่มีการจัดการข้อมูลที่ยุ่งยาก
2) สร้างผังโครงสร้างเมนู (Menu Structure) ที่เข้าถึงแบบจำลองที่ได้เลือกในข้อ 1 โดยในผังโครงสร้างนี้จะประกอบด้วยเมนู และส่วนติดต่อกับผู้ใช้ (Dialogue)
3) สร้างเส้นทางเข้าถึงส่วนติดต่อกับผู้ใช้แบบจำลอง (Pathway)
4) สร้างส่วนประกอบของจอภาพ โดยใช้แผนผังโครงสร้างอินพุต/เอาท์พุต (I/O Structure)
5) จัดกลุ่มของสมาชิกภายในผังโครงสร้างอินพุต/เอาท์พุต
6) สร้างเส้นทางของแบบจำลองจากเมนูถึงส่วนประกอบของจอภาพและรายงาน
7) พัฒนาแบบจำลองที่ได้มีการสร้างเส้นทาง
8) แสดงแบบจำลองให้ผู้ใช้ดูเพื่อสอบถาม ขั้นตอนบางอย่างและให้ผู้ใช้ตรวจสอบแบบจำลองที่ได้ โดยในการสอบถามอาจจัดทำเอกสารในการสอบถาม
9) นำข้อมูลที่ได้จากผู้ใช้มาปรับปรุงแบบจำลอง
10) จัดทำรายงานของแบบจำลอง

รูป 2.41 แสดงขั้นตอนการพัฒนาแบบจำลองข้อกำหนดเฉพาะ

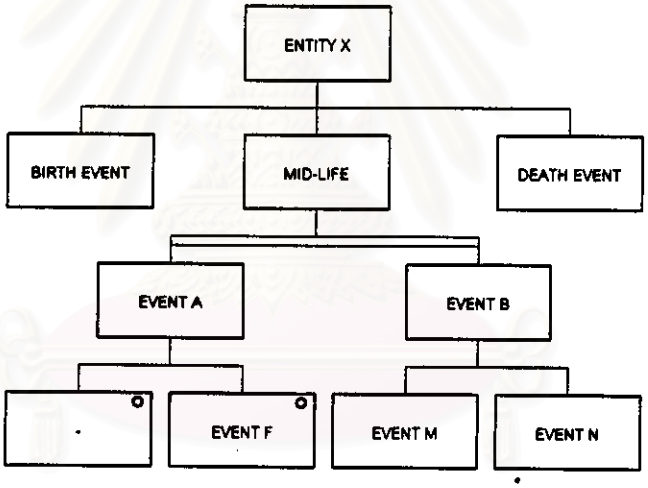
2.2.8 แบบจำลองเอนทิตี้-เหตุการณ์ (Entity-Event Modelling) ในการสร้างแบบจำลองของกระแสข้อมูลและการนิยามฟังก์ชัน เป็นการสร้างมุมมองของขั้นตอนการทำงานในระดับบน แต่การสร้างแบบจำลองข้อมูลเชิงตรรกและการวิเคราะห์ข้อมูลเชิงสัมพันธ์จะทำให้ได้โครงสร้างของข้อมูลสำหรับระบบใหม่ แต่แบบจำลองเอนทิตี้-เหตุการณ์เป็นการนำทั้งขั้นตอนการดำเนินงาน และโครงสร้างของข้อมูลมาศึกษาถึงผลกระทบของข้อมูล เมื่อเวลาเปลี่ยนแปลง แบบจำลองนี้จะใช้เทคนิค 2 วิธี คือ การวิเคราะห์ประวัติชีวิตของเอนทิตี้ (Entity Life History, ELH) และแผนภาพความเกี่ยวพันของผลกระทบที่ได้รับ (Effect Correspondence Diagram, ECD)

2.2.8.1 การวิเคราะห์ประวัติของเอนทิตี้ (Entity Life History, ELH) การวิเคราะห์ประวัติของเอนทิตี้เป็นการศึกษาถึงวงจรชีวิตของเอนทิตี้ ตั้งแต่เริ่มสร้างไปจนถึงการสิ้นสุดของแต่ละเอนทิตี้ โดยมีเหตุการณ์และผลกระทบที่ได้รับจากเหตุการณ์ที่เกิดขึ้น สัญลักษณ์ที่ใช้ในแผนภาพวิเคราะห์ประวัติชีวิตของเอนทิตี้จะใช้สัญลักษณ์เดียวกับผังงานโครงสร้างของแจ็คสัน (Jackson-like Structure)

หรือผังโครงสร้างอินพุต/เอาท์พุต (I/O Structure) ตามรูปที่ 2.42 บางครั้งไม่สามารถที่จะทราบได้ว่าเหตุการณ์ใดเกิดก่อน สามารถใช้ผังโครงสร้างขนาน (Parallel Structure) ได้ ตามรูป 2.43 สำหรับขั้นตอนการพัฒนาแผนภาพประวัติชีวิตของเอนทิตี้ แสดงตามรูป 2.44



รูป 2.42 แสดงแผนภาพประวัติชีวิตของเอนทิตี้



รูป 2.43 แสดงผังโครงสร้างขนาน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ขั้นตอนการพัฒนาแผนภาพประวัติชีวิตของเอนทิตี้

1) เขียนเหตุการณ์ทั้งหมดที่มีผลกระทบกับแต่ละเอนทิตี้ ในรูปของตารางเหตุการณ์/เอนทิตี้ (Event/Entity Matrix) ดูว่าเหตุการณ์ใดที่สร้าง (Creation, C), ลบ (Deletion, D), หรือเปลี่ยนแปลง (Modification, M) แต่ละเอนทิตี้
2) เขียนแผนภาพประวัติของเอนทิตี้อย่างง่าย โดยเริ่มจากเหตุการณ์สร้างเอนทิตี้ที่เปลี่ยนแปลงและที่ลบ
3) ตรวจสอบแผนภาพที่สร้างขึ้นว่าแต่ละเอนทิตี้ มีความสัมพันธ์กับเอนทิตี้อื่นหรือไม่ เช่น เมื่อมีการลบเอนทิตี้หลัก (Master Entity) มีผลทำให้เอนทิตี้ย่อยถูกลบด้วยหรือไม่, เมื่อเอนทิตี้ย่อยถูกลบทั้งหมดมีผลให้เอนทิตี้หลักถูกลบด้วยหรือไม่ หรือเอนทิตี้หลักต้องถูกลบได้ก็ต่อเมื่อไม่มีเอนทิตี้ย่อยอยู่แล้ว เป็นต้น
4) กำหนดการดำเนินงาน (Operations) ทางด้านตรรก ซึ่งเป็นการอธิบายขั้นตอนการทำงานตามลักษณะเฉพาะและความสัมพันธ์ของเอนทิตี้ที่ด้านล่างของแต่ละเหตุการณ์ที่เกิดขึ้น

รูป 2.44 แสดงขั้นตอนการพัฒนาแผนภาพประวัติชีวิตของเอนทิตี้

2.2.8.2 แผนภาพความเกี่ยวพันของผลกระทบ (Effect Correspondence Diagram, ECDs) เป็นการศึกษาถึงผลกระทบที่เกิดขึ้นกับหลาย ๆ เอนทิตี้ จากมุมมองของเหตุการณ์ใด ๆ โดยมีขั้นตอนการพัฒนาแผนภาพความเกี่ยวพันของผลกระทบตามรูป 2.45

ขั้นตอนการพัฒนาแผนภาพความเกี่ยวพันของผลกระทบ

1) เลือกทุก ๆ เอนทิตี้ที่มีผลกระทบต่อเหตุการณ์ แล้วนำมาวาดในรูปของสี่เหลี่ยมที่มีมุมมนโดยดูจากแผนภาพข้อมูลเชิงตรรก
2) เลือกผลกระทบของเหตุการณ์ที่เกิดขึ้นในเวลาเดียวกัน ซึ่งอาจจะเกิดหรือไม่เกิดก็ได้
3) เขียนเหตุการณ์ที่มีผลกระทบกับเอนทิตี้ที่เป็นทางเลือกโดยก่อนเขียนผลกระทบที่เป็นทางเลือกที่ด้านล่างของเอนทิตี้
4) เขียนเหตุการณ์ที่มีผลกระทบกับเอนทิตี้ที่ซ้ำ (วนลูป) โดยการเขียนผลกระทบที่เป็นวนลูปที่ด้านล่างของเอนทิตี้
5) เลือกผลกระทบที่มีความเกี่ยวพันแบบ 1 ต่อ 1 และเชื่อมเอนทิตี้ที่มีความเกี่ยวพันกันด้วยเส้นที่ปลายเส้นมีหัวลูกศร 2 ด้าน
6) บางเอนทิตี้อาจได้รับผลกระทบที่ซ้ำ (วนลูป) มากกว่า 1 ลูป อาจจะเขียนโครงสร้างที่ซ้ำโดยต้องเชื่อมกับการวนซ้ำของทางเลือกแต่ละแบบ หรือทางเลือกของการวนซ้ำ
7) เพิ่มเอนทิตี้อื่นที่มีความสัมพันธ์/ความเกี่ยวพัน การวนซ้ำของทางเลือกแต่ละแบบ
8) แสดงรายการข้อมูลรับเข้าภายใต้ทางเข้า (Entry Point) สำหรับเหตุการณ์

รูป 2.45 แสดงขั้นตอนการพัฒนาแผนภาพความเกี่ยวพันของผลประทบ

2.2.9 การออกแบบการดำเนินงานของฐานข้อมูลเชิงตรรก (Logical Database Process Design, LDPD) จะใช้ในขั้นตอนเลือกแบบจำลองในส่วนปรับปรุง (Update Process Models) และในส่วนการสอบถาม (Enquiry Process Models) เป็นเทคนิคการออกแบบโปรแกรมมาตรฐานของการรวมแผนผังโครงสร้างข้อมูลอินพุตและเอาท์พุต ให้เป็นผังโครงสร้างของการดำเนินงาน สัญลักษณ์ที่ใช้ในแผนภาพแบบจำลองโปรเซสการปรับปรุง หรือแบบจำลองการสอบถามจะใช้สัญลักษณ์เดียวกับผังงานโครงสร้างของแจ็คสัน หรือผังโครงสร้างอินพุต/เอาท์พุตและแผนภาพวิเคราะห์ประวัติของเอนทิตี้ ขั้นตอนการสร้างแบบจำลองในส่วนงานปรับปรุงจะแสดงตามรูป 2.46 และขั้นตอนการสร้างแบบจำลองในส่วนงานสอบถาม จะแสดงตามรูป 2.47 ในส่วนการสร้างแบบจำลองในส่วนงานสอบถามนั้นต้องอาศัยแผนภาพเส้นทางการเข้าถึงการสอบถาม (Enquiry Access Path, EAP) เพื่อใช้ในการศึกษาถึงเส้นทางการเข้าถึงของส่วนงานการสอบถามข้อมูล

ขั้นตอนการสร้างแบบจำลองในส่วนงานปรับปรุง

1) สร้างผังโครงสร้างเอาท์พุต (Output Structure) โดยนำมาจากผังงานอินพุต/ เอาท์พุต ที่ได้จากเสต็ป 330 โดยการลบสมาชิกที่เป็นอินพุตออกจากโครงสร้าง
2) สร้างผังโครงสร้างอินพุต (Input Structure) โดยนำมาจากแผนภาพความเกี่ยวพันของผลกระทบสำหรับเหตุการณ์เดียวกับข้อ 1 มาดูเพิ่มเอนทิตี้ที่มีความสัมพันธ์กับเหตุการณ์
3) จัดกลุ่มผลกระทบที่มีความสัมพันธ์แบบ 1 ต่อ 1
4) ทำการเปลี่ยนแผนภาพความเกี่ยวกันของผลกระทบให้เป็นผังโครงสร้างอินพุต/เอาท์พุต จะได้ผังโครงสร้างอินพุต
5) นำแผนผังอินพุตและเอาท์พุตมาเขียนด้วยกัน และหาความสัมพันธ์/ความเกี่ยวข้องของแต่ละสมาชิกในผังอินพุตและเอาท์พุต
6) ทำการเชื่อมผังโครงสร้างอินพุตและเอาท์พุต ให้เป็นผังโครงสร้างการดำเนินงาน
7) กำหนดการดำเนินงาน (Operation) เช่น การอ่าน (Read) ของข้อมูล, เขียน (Write) ข้อมูล หรืออาจกำหนดเงื่อนไขข้อผิดพลาดที่เกิดขึ้นจากการดำเนินงาน รวมถึงกำหนดว่าเมื่อมีข้อผิดพลาดเกิดขึ้นข้อความที่จะปรากฏคืออะไร

รูป 2.46 แสดงขั้นตอนการสร้างแบบจำลองในส่วนงานปรับปรุง

ขั้นตอนการสร้างแบบจำลองในส่วนงานสอบถาม

1) สร้างผังโครงสร้างเอาท์พุต (Output Structure) โดยนำมาจากผังงานอินพุต/ เอาท์พุต ที่ได้จากเสตีป 330 โดยการลบสมาชิกที่เป็นอินพุตออกจากผังโครงสร้าง
2) สร้างผังโครงสร้างอินพุต (Input Structure) โดยนำมาจากแผนภาพเส้นทางการเข้าถึงการสอบถาม (Enquiry Access Paths) สำหรับเหตุการณ์เดียวกับข้อ 1 สำหรับขั้นตอนการสร้างแผนภาพนี้แสดงตามรูป 2.48
3) จัดกลุ่มการเข้าถึงการสอบถามที่มีความสัมพันธ์แบบ 1 ต่อ 1
4) ทำการเปลี่ยนแผนภาพการเข้าถึงการสอบถามให้เป็นผังโครงสร้างอินพุต/ เอาท์พุต จะทำให้ได้ผังโครงสร้างอินพุต
5) นำแผนผังอินพุตและเอาท์พุตมาเขียนเปรียบเทียบกัน และหาความสัมพันธ์/ ความเกี่ยวข้องของแต่ละสมาชิกในผังอินพุตและเอาท์พุต
6) ทำการเชื่อมผังโครงสร้างอินพุตและเอาท์พุต ให้เป็นผังโครงสร้างการดำเนินงาน
7) กำหนดการดำเนินงาน (Operation) เช่น การอ่านคีย์ของข้อมูล (Read by key), การอ่านข้อมูลหลัก ตามไปถึงกำหนดว่าเมื่อมีข้อผิดพลาดที่เกิดจากเงื่อนไขและผลลัพธ์ที่ได้กำหนดเงื่อนไขและผลลัพธ์ที่เกิดข้อผิดพลาด

รูป 2.47 แสดงขั้นตอนการสร้างแบบจำลองในส่วนการสอบถาม

ขั้นตอนการสร้างแผนภาพเส้นทางการเข้าถึงการสอบถาม

1) ต้องกำหนดถึงชื่อของการสอบถาม
2) กำหนดกฎเกณฑ์ของการสอบถาม ประกอบด้วยคีย์ที่ใช้ในการสอบถามข้อมูล
3) เลือกเอนทิตี้ที่ใช้ในการสอบถามข้อมูล
4) วาดความสัมพันธ์ของแต่ละเอนทิตี้ที่เกี่ยวข้องตามระบบที่ต้องการ ซึ่งใช้สัญลักษณ์ความสัมพันธ์ของแบบจำลองข้อมูลเชิงตรรก
5) วาดแผนภาพเส้นทางการเข้าถึงการสอบถาม โดยการเขียนสัญลักษณ์ที่ใช้แทนเหตุการณ์ที่เป็นทางเลือก หรือการวาดรูป รวมทั้งเชื่อมเอนทิตี้ที่มีความเกี่ยวพันกันด้วย หัวลูกศรที่ปลายด้านเดียว
6) เขียนคุณสมบัติของคีย์ที่ใช้ในการค้นหา/สอบถามข้อมูลที่ทางเข้า (Entry Point) เช่นใช้คีย์นัลหรือลักษณะเฉพาะอื่น
7) ตรวจสอบว่าการอ่านซึ่งเป็นการดำเนินงานที่ใช้ (Operations) สำหรับการสอบถามในแต่ละเอนทิตี้ใช้คีย์อะไรในการอ่านข้อมูล

รูป 2.48 แสดงขั้นตอนการสร้างแผนภาพเส้นทางการเข้าถึงการสอบถาม

2.2.10 การออกแบบข้อมูลทางกายภาพ (Physical Data Design) แผนภาพกระแสข้อมูลเชิงตรรกของระบบที่ต้องการให้ภาพในมุมมองของข้อมูลของระบบตามความต้องการของผู้ใช้ แต่ไม่ได้บอกถึงการจัดการหรือเก็บข้อมูลทางกายภาพทำอย่างไร โดยการออกแบบข้อมูลทางกายภาพจะเป็นการ

แปลงกระแสข้อมูลเชิงตรรกของระบบที่ต้องการให้อยู่ในรูปการออกแบบข้อมูลทางกายภาพ ซึ่งมีขั้นตอนที่ทำ 2 ขั้นตอน คือ การออกแบบข้อมูลเบื้องต้น (First-cut Data Design) ซึ่งในขั้นตอนนี้จะสร้างการออกแบบข้อมูลที่คล้ายกับแผนภาพกระแสข้อมูลเชิงตรรก มีขั้นตอนการออกแบบข้อมูลเบื้องต้นตามรูป 2.49 และการเพิ่มประสิทธิภาพของการออกแบบข้อมูล (Optimized Data Design) เป็นการปรับการออกแบบข้อมูลเบื้องต้นโดยเน้นประสิทธิภาพการทำงาน มีขั้นตอนการออกแบบการเพิ่มประสิทธิภาพของการออกแบบตามรูป 2.50

ขั้นตอนการออกแบบข้อมูลเบื้องต้น

1) เปลี่ยนแปลง/ปรับแผนภาพกระแสข้อมูลเชิงตรรก โดยการเปลี่ยนรูปสี่เหลี่ยมมุมมนเป็นรูปสี่เหลี่ยมจัตุรัส
2) ความสัมพันธ์ระหว่างเอนทิตี้หลักกับเอนทิตี้ย่อยที่เป็นทางเลือก (Optionality) ให้ใช้ความสัมพันธ์เส้นหนาที่มีตัวอักษร "O" อยู่บนกึ่งกลางเส้นตามรูป 2.51
3) ความสัมพันธ์ระหว่างเอนทิตี้หลักกับเอนทิตี้ย่อยที่เป็นส่วนโค้งพิเศษ (Exclusivety Arcs) ถูกแทนด้วยความสัมพันธ์ใช้เส้นหนาที่ตัวอักษร "O" อยู่บนกึ่งกลางเส้นตามรูปที่ 2.52
4) เขียนปริมาตรหรือขนาดของข้อมูล (จำนวนระเบียน) ของแต่ละเอนทิตี้ลงในรูปสี่เหลี่ยมที่ใช้แผนเอนทิตี้ลงในแผนภาพ และอัตราส่วนของขนาดข้อมูลระหว่างเอนทิตี้หลักกับเอนทิตี้ย่อยลงบนเส้นความสัมพันธ์
5) เขียนเพิ่มจุดทางเข้าที่ต้องการ โดยอาศัยจุดทางเข้าของแผนภาพ เส้นทางการเข้าถึงการสอบถาม (EAPs) และแผนภาพความเกี่ยวพันของผลกระทบ (ECDs) โดยสัญลักษณ์ที่ใช้เป็นลูกศรข้างใดข้างหนึ่ง พร้อมด้วยข้อมูลที่ใช้ในการเข้าถึง
6) เลือกเอนทิตี้ที่เป็นส่วนบนสุดของการเข้าถึงกลุ่มของเอนทิตี้หลัก-เอนทิตี้ย่อย โดยรากคือเอนทิตี้ที่ไม่มีเอนทิตี้หลักตัวอื่นอยู่เหนือลำดับขั้น โดยสัญลักษณ์ที่ใช้เป็นการขีดเส้นบนเหนือเอนทิตี้หลัก
7) จัดกลุ่มของเอนทิตี้ย่อยเข้ากลุ่มกับเอนทิตี้หลัก โดยการวงกลมล้อมรอบกลุ่มของเอนทิตี้ที่อยู่ในกลุ่มเดียวกัน
8) บางครั้งเอนทิตี้ย่อยอาจถูกจัดกลุ่มมากกว่า 1 กลุ่ม จะต้องเลือกเพียง 1 กลุ่มเท่านั้น

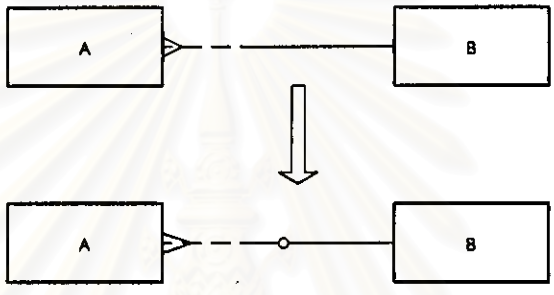
รูป 2.49 แสดงขั้นตอนการออกแบบข้อมูลเบื้องต้น

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

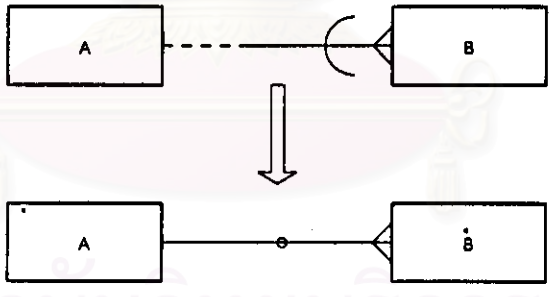
ขั้นตอนการเพิ่มประสิทธิภาพการออกแบบข้อมูล

1) ศึกษาถึงวัตถุประสงค์ที่กำหนดจากรายการความต้องการเพื่อศึกษาผลกระทบกับการออกแบบข้อมูลกายภาพ
2) คำนวณขนาดของพื้นที่ใช้เก็บข้อมูลโดยการคำนวณขนาดของทุก ๆ เอนทิตี้ นอกจากนี้ต้องคิดถึงขนาดของพื้นที่เก็บเอนทิตี้ (Indexes), พอยน์เตอร์ (Pointer)
3) ให้ทำการปรับโครงสร้างของข้อมูลโดยอาจเปรียบเทียบเพื่อลดขนาดของเวลาที่ใช้ในการเข้าถึงข้อมูล รวมถึงศึกษาถึงสิ่งอำนวยความสะดวกที่ระบบจัดการฐานข้อมูลมีให้

รูป 2.50 แสดงขั้นตอนการเพิ่มแระสิทธิภาพการออกแบบข้อมูล



รูป 2.51 แสดงการเปลี่ยนรูปความสัมพันธ์ที่เป็นทางเลือก



รูป 2.52 แสดงการเปลี่ยนรูปความสัมพันธ์ที่เป็นส่วนโค้งพิเศษ

2.3 ผลลัพธ์ที่เกิดขึ้น

ในแต่ละเสต็ปจะได้ผลลัพธ์ออกมาตามตารางที่แสดงในกรอบขั้นตอนการทำงาน โดยแบ่งผลลัพธ์ได้เป็น 3 กลุ่มใหญ่ ๆ คือ กลุ่มแรกคือขั้นตอนการดำเนินงาน (Processing) ประกอบไปด้วย แผนภาพกระแสข้อมูลทั้งหมดที่ได้จากแบบจำลองกระแสข้อมูล การนิยามฟังก์ชัน แบบจำลองเอนทิตี้กับเหตุการณ์ การออกแบบการออกแบบการดำเนินงานของฐานข้อมูลเชิงตรรก แผนผังสำหรับการพัฒนา

ส่วนประกอบฟังก์ชัน และข้อกำหนดเฉพาะของโปรแกรม กลุ่มที่สองคือเรื่องข้อมูล (Data) ประกอบไปด้วย แผนภาพข้อมูลเชิงตรรกที่ได้จากการแบบจำลองข้อมูลเชิงตรรก การวิเคราะห์ข้อมูลเชิงสัมพันธ์ การวิเคราะห์ประวัติชีวิตเอนทิตี้ และการออกแบบข้อมูลทางกายภาพ กลุ่มสุดท้ายส่วนติดต่อระหว่างระบบ-ผู้ใช้ (System-User Interface) ประกอบไปด้วย การกำหนดส่วนติดต่อกับผู้ใช้ แผนผังโครงสร้างอินพุต-เอาท์พุต และแผนผังโครงสร้างการติดต่อกับผู้ใช้ โดยแต่ละแผนภาพหรือผลลัพธ์ที่เกิดขึ้นอาจต้องใช้ขั้นตอนการสร้างหลายเสต็ป เช่น แบบจำลองกระแสข้อมูล จะถูกสร้างทั้งเฟส 1 ศึกษาความเป็นไปได้ของระบบ เฟส 2 การวิเคราะห์ความต้องการของระบบ และเฟส 3 ข้อกำหนดเฉพาะของระบบที่ต้องการ เป็นต้น

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย