



ปีการศึกษา 2532

การพัฒนาระบบงานคอมพิวเตอร์สำหรับภาควิชาวิศวกรรมคอมพิวเตอร์
MIS FOR COMPUTER ENGINEERING DEPARTMENT

จัดทำโดย

รัตนะ รัตนพันธ์

อาจารย์ที่ปรึกษา

อ.วิษระ ฉัตรวิริยะ



ปฏิทินนิพนธ์ปีการศึกษา 2532

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การพัฒนาระบบงานคอมพิวเตอร์สำหรับภาควิชาวิศวกรรมคอมพิวเตอร์

MIS FOR COMPUTER ENGINEERING DEPARTMENT

ผู้จัดทำ รัตนะ รัตนพันธ์

..... อาจารย์ที่ปรึกษา
(อ.วิษยะ ฉัตรวิริยะ)

การพัฒนาระบบงานคอมพิวเตอร์สำหรับภาควิชาวิศวกรรมคอมพิวเตอร์
MIS FOR COMPUTER ENGINEERING DEPARTMENT

จัดทำโดย

รัตนะ รัตนพันธ์

Ratana Ratanapan

อาจารย์ที่ปรึกษา

อ. วีระ ฉัตรวิริยะ

Watchara Chatwiriya

ปริญญาโท สาขาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2532

การพัฒนาระบบงานคอมพิวเตอร์สำหรับภาควิชาวิศวกรรมคอมพิวเตอร์

วิคนะ รัตนพันธ์

อ.วิษยะ จิตรวิษยะ อาจารย์ที่ปรึกษา
ปีการศึกษา 2532

บทคัดย่อ

ปฏิญานินตฉบับนี้ เป็นการนำระบบสารสนเทศมาประยุกต์ใช้ในภาควิชาภาควิชาวิศวกรรมคอมพิวเตอร์ ระดับแผนก ภาควิชาวิศวกรรมคอมพิวเตอร์ เริ่มต้นจากการวิเคราะห์และออกแบบฐานข้อมูลจากข้อมูลที่เกิดขึ้นจริงๆ ก็จะได้ขั้นตอนการดำเนินงานต่างๆจาก ไอแซค(ISAC) แล้วจึงใช้ในแอม(NIAM) หากความสัมพันธ์ระหว่างข้อมูลเหล่านั้น แล้วนำมาเขียนแผนภาพ ในแอม แสดงความสัมพันธ์ระหว่างข้อมูล แล้วนำผลที่ได้ขึ้นมาแบ่งแยกเป็นตาราง

เมื่อได้ตารางครบแล้ว ก็ถึงขั้นตอนการโปรแกรมซึ่งแบ่งเป็น 2 ส่วนคือ ส่วนที่แสดงผลและส่วนระบบฐานข้อมูล โดยส่วนแสดงผลจะเน้นถึงความสะดวกและใช้งานได้ง่ายสำหรับผู้ใช้ซึ่งในขั้นนี้ ได้จำลองรูปแบบการทำงานบนไมโครคอมพิวเตอร์ เมื่อได้ผลตามที่ต้องการแล้วจึงเปลี่ยนมาลงบนยูนิกซ์ ส่วนระบบฐานข้อมูลก็ใช้โปรซี(PRO* C) ซึ่งเป็นคอมไพล์เลอร์ที่จะเปลี่ยนจากเอสคิวแอล(SQL)ให้เป็นภาษาซี แล้วจึงนำทั้งสองส่วนมารวมกัน โดยโปรแกรมทั้งหมดจะถูกแบ่งออกเป็นโปรแกรมย่อยที่เป็นอิสระต่อกัน แต่สามารถเรียกนำมาใช้ร่วมกันได้ โดยใช้ซิสเต็มคอล (SYSTEM CALL)

MIS FOR COMPUTER ENGINEERING DEPARTMENT

Ratana Ratanapan

Watchara Chatviriya Advisor

1989

Abstract

This thesis is an application of MIS in governmental section such as Computer Engineering Department. Beginning with database analysis and design on authentic data that get step of operation from ISAC. Then using NIAM searches the relation of all those data. Consequently this result use for drawing NIAM graph which demonstrate each relation. After that NIAM will divide all relation to be table.

When all tables were created, next step is programming. This program has two parts. First is display and another one is DBMS. Display part emphasis on convenience and easy usage. This step simulate principal function on microcomputer. When every function succeed, transferring display program on UNIX. For DBMS part we use PRO*C for precompiling from SQL to C Language. Furthermore both two parts are joined together. Main program will be divided to many independent subprogram but every subprogram can call from main program by using System Call.

สารบัญ

		หน้า
บทที่ 1	บทนำ	1
บทที่ 2	ทฤษฎีและหลักการ	2
บทที่ 3	การสร้างโปรแกรม	47
บทที่ 4	ผลการวิเคราะห์	53
บทที่ 5	สรุปวิจารณ์	68
ภาคผนวก ก	Rand Editor	69
ภาคผนวก ข	Screenlib	75
ภาคผนวก ค	ทฤษฎีไอแซค (ISAC)	80
ภาคผนวก ง	ไนแอม (NIAM)	105
ภาคผนวก จ	เอ-กราฟ (A-GRAPH)	111
ภาคผนวก ฉ	เอสคิวแอลพลัส (SQL*PLUS)	120
ภาคผนวก ช	โปรซี (PRO*C)	143
ภาคผนวก ซ	การใช้เมคไฟล์ (Makefile)	145
ภาคผนวก ฅ	การเรียกใช้เอสคิวแอลจากภาษาสูง	148
กิตติกรรมประกาศ		153
หนังสืออ้างอิง		154

บทที่ 1

บทนำ

ในปัจจุบันเนื่องมาจากการพัฒนาทางเทคโนโลยีแขนงต่างๆ ที่เพิ่มมากขึ้น เป็นผลให้มีการใช้ข้อมูลประเภทต่างๆ เป็นจำนวนมาก ซึ่งทำให้เทคโนโลยีสาขาหนึ่งได้รับการสนใจ และพัฒนาควบคู่กันมาด้วย คือ เทคโนโลยีสารสนเทศ (Information Technology) ซึ่งเกี่ยวข้องกับการพัฒนาฐานข้อมูลและระบบจัดการฐานข้อมูล

ภาควิชาวิศวกรรมคอมพิวเตอร์ก็เป็นอีกหน่วยงานหนึ่ง ที่ต้องการเทคโนโลยีใหม่ๆ มาใช้ในการบริการด้านข้อมูลแก่บุคลากรของภาค นักศึกษา และบุคคลภายนอก การที่จะสามารถตอบสนองความต้องการต่างๆ ได้อย่างครบถ้วน ก็จะต้องแบ่งการทำงานออกเป็นขั้นตอน ดังนี้

ขั้นตอนการดำเนินงาน

1. ศึกษาความต้องการของบุคลากรที่เกี่ยวข้อง โดยการออกแบบสอบถามและการสัมภาษณ์ แล้วรวบรวมข้อมูลทุกอย่างมาวิเคราะห์ หาข้อสรุป
2. นำข้อมูลที่ได้วิเคราะห์ได้มาศึกษา หาความเป็นไปได้ที่จะนำคอมพิวเตอร์มาช่วยในการทำงาน โดยคำนึงถึงประโยชน์ที่จะได้รับเพิ่มขึ้น จากระบบเดิมกับระบบใหม่ที่จะพัฒนาขึ้น
3. วิเคราะห์ข้อมูลทั้งหมด แล้วออกแบบฐานข้อมูล ให้มีประสิทธิภาพ โดยแบ่งแยกข้อมูลที่มีความสัมพันธ์กันออกเป็นชุด ซึ่งก็จะได้ข้อมูลออกมาเป็นตาราง อย่างครบถ้วน
4. ออกแบบหน้าจอ โดยคำนึงถึงความสะดวกต่อการใช้งาน โดยใช้ภาษา C จำลองการใช้งานจริง ๆ ซึ่งจะมีการกำหนดขั้นตอนลำดับการทำงานของโปรแกรม
5. ศึกษาการทำงานของระบบยูนิกซ์ แล้วย้ายโปรแกรมหน้าจอ ลงบนยูนิกซ์ โดยเปลี่ยนแปลงนึ่งก็ชั่น และคำสั่งต่างๆ ให้เหมาะสม
6. เขียนโปรแกรม SQL*Plus จัดการฐานข้อมูล ประกอบด้วยการสร้างตาราง เพิ่มเติมข้อมูล ลบข้อมูล แก้ไขข้อมูล และค้นหาข้อมูล
7. นำโปรแกรมทั้งสองส่วนมารวมกัน โดยใช้ PRO*C ซึ่งเป็นตัวที่จะคอมไพล์จาก SQL ให้เป็นภาษา C
8. ทดสอบ แก้ไข ปรับปรุง การทำงานของโปรแกรม และจัดทำคู่มือการใช้งาน

บทนิยาม

บทบัญญัติ

ในบทนี้จะกล่าวถึงทฤษฎีต่างๆที่เกี่ยวข้องกับการทำงานในโครงการดังนี้

2.1 การพัฒนาระบบสารสนเทศโดยวิธีการของไอแอล : จะกล่าวถึงรายละเอียดขั้นตอนต่างๆในการพัฒนาระบบสารสนเทศ ทั้งส่วนที่ใช้ในโครงการและบางส่วนที่ไม่ได้ใช้ โดยแบ่งหัวข้อเป็น

2.1.1 ความเป็นมาและลักษณะวิธีการของไอแอล

2.1.2 แอคทีวิตี้ กราฟ

2.1.3 การวิเคราะห์การเปลี่ยนแปลง

2.2 ระบบฐานข้อมูล: จะกล่าวถึงรายละเอียดต่างๆในเรื่องของระบบฐานข้อมูล แบ่งเป็นส่วนต่างๆดังนี้

2.2.1 ความหมายและที่มา

2.2.2 ระบบจัดการฐานข้อมูล

2.2.3 โมเดลของข้อมูล

2.2.4 สถาปัตยกรรมของระบบฐานข้อมูล

2.2.5 ผู้ใช้

2.3 ระบบจัดการฐานข้อมูลแบบสัมพันธ์ ออเรเคิล : จะกล่าวถึงรายละเอียดทั่วไปของออเรเคิล และคุณสมบัติในฐานะที่เป็นระบบจัดการฐานข้อมูลแบบสัมพันธ์ แบ่งหัวข้อดังนี้

2.3.1 ออเรเคิล

2.3.2 การพิจารณาคุณลักษณะของออเรเคิลในฐานะระบบจัดการฐานข้อมูลแบบสัมพันธ์

2.4 การออกแบบฐานข้อมูลแบบสัมพันธ์ในระดับแนวคิดโดยอาศัยโนแอมโมเดล กล่าวถึงการออกแบบฐานข้อมูล รายละเอียดต่างของโนแอม และการออกแบบฐานข้อมูลโดยโนแอม แบ่งหัวข้อเป็น

2.4.1 การออกแบบฐานข้อมูล

2.4.2 โมเดลของวิธีการวิเคราะห์ข้อมูลของโนแอม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 การศึกษาระบบสารสนเทศโดยวิธีการของไอเอส

2.1.1 ความเป็นมาและลักษณะวิธีการของไอเอส

ระบบสารสนเทศ (Information System)

เราสามารถพิจารณาจากระบบสารสนเทศในแง่ของความร่วมมือระหว่างกลุ่มบุคคลในการจัดการและส่งผ่านข้อมูล (Information) ซึ่งอาจมีการนำคอมพิวเตอร์มาช่วยงานด้วย ดังนั้นเราสามารถให้คำจำกัดความระบบสารสนเทศได้ว่า ระบบสารสนเทศได้แก่ระบบซึ่งพัฒนาขึ้นเพื่อสร้าง (create), เก็บรวบรวม (collect), เก็บรักษา (store), ทำกรรมวิธี (process), กระจาย (distribute) และแปลความหมาย (interpret) เซตของข้อมูล (Information Set) [หนังสืออ้างอิง 1] ระบบสารสนเทศจะต้องถูกพัฒนาขึ้นก่อนที่จะนำมาใช้งาน

การพัฒนาสารสนเทศ (Information Systems Development) ประกอบด้วย

- การวิเคราะห์ระบบสารสนเทศ (Analysis)
- การออกแบบระบบสารสนเทศ (Design)
- การสร้างระบบสารสนเทศ (Realization)

ในระบบสารสนเทศจะประกอบไปด้วยกิจกรรมหรือขั้นตอนทั้งที่กระทำโดยมนุษย์ คอมพิวเตอร์ หรืออุปกรณ์ช่วยอื่นๆ ซึ่ง กิจกรรมที่กระทำโดยมนุษย์ได้แก่การล้สร้าง การแปลความหมายข้อมูล ส่วนที่กระทำโดยคอมพิวเตอร์หรืออุปกรณ์ช่วยอื่นๆนั้น ได้แก่กิจกรรมที่สามารถทำในลักษณะอัตโนมัติ (automatable) ระบบสารสนเทศถูกสร้างมาเพื่อจุดประสงค์ให้คนสามารถทำงานได้อย่างมีประสิทธิภาพมากขึ้น ดังนั้นระบบสารสนเทศจึงเป็นเพียงเครื่องมืออย่างหนึ่งเท่านั้นซึ่งผู้ใช้จะต้องความคุมและรับผิดชอบในการใช้เครื่องมืออันนี้ การใช้ระบบสารสนเทศให้เกิดประสิทธิภาพจึงขึ้นอยู่กับตัวผู้ใช้เป็นสำคัญว่ามีความเข้าใจจุดมุ่งหมายของตนมากเพียงใด การใช้ระบบสารสนเทศอย่างไม่พิจารณาให้ถี่ถ้วนนอกจากจะไม่เกิดประโยชน์แล้ว ยังจะก่อให้เกิดผลในแง่ลบอีกด้วย

ความเป็นมาของไอเอส

วิธีการในการพัฒนาระบบสารสนเทศมีอยู่ด้วยกันมากมายหลายรูปแบบหลายวิธีการ การพัฒนาระบบสารสนเทศตามวิธีของไอเอสก็เป็นหนึ่งในวิธีการเหล่านั้น

ไอเอสได้ถูกวิจัยและคิดค้นขึ้นโดยกลุ่มนักวิจัยประจำภาควิชา Administrative Information Processing แห่งสถาบัน Royal Institute of Technology และ University of Stockholm ประเทศสวีเดน ตั้งแต่ปี ค.ศ. 1971 ในระหว่างการวิจัยนั้น ไอเอสได้ถูกนำไปทดลองใช้ในบริษัทธุรกิจหลายประเภท และในองค์กรต่างๆ ซึ่งเป็นผลทำให้เกิดการพัฒนาหลักวิธีการต่างๆมากมายกับไอเอส และทำให้มีการนำไอเอสไปใช้อย่างแพร่หลายในวง การธุรกิจและการศึกษาในประเทศกลุ่มสแกนดิเนเวีย [หนังสืออ้างอิง 1] ในปี ค.ศ. 1981 ได้มีเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้เผยแพร่เป็นการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตั้งสถาบันชื่อ Institute for Development of Activities in Organization
ขึ้นในประเทศไทยเมื่อทำการศึกษาวิจัยเพิ่มเติมในเรื่องของไอแอลขึ้นด้วย

ขั้นตอนวิธีการของไอแอล

จากรูปที่ 2.1 แสดงให้เห็นขั้นตอนต่างๆในการพัฒนาระบบ ขั้นตอนในการพัฒนาระบบทั้ง
กล่าวภายใต้วิธีการของไอแอลมีดังต่อไปนี้

วิเคราะห์การเปลี่ยนแปลง (Change Analysis) จะเห็นได้ว่าขั้นตอนแรกก่อนที่จะทำ
การพัฒนาระบบสารสนเทศก็คือการ การวิเคราะห์การเปลี่ยนแปลงจะเป็นตัวบ่งชี้หรือนำไปสู่ข้อ
สรุปว่า การนำระบบสารสนเทศไปใช้หรือการพัฒนาระบบสารสนเทศจะสามารถแก้ปัญหา หรือ
สนองความต้องการหรือไม่? หรือว่าควรจะหันมาในด้านอื่นแทน? มีปัญหาบางประเภทที่การพัฒนาระบบ
สารสนเทศไม่สามารถแก้ไขได้ ดังนั้นเราจะต้องใช้การพัฒนาทางด้านอื่นแทน ตัวอย่างเช่น
การพัฒนาองค์การ(Organization Development) การพัฒนาความสัมพันธ์ของบุคลากรใน
หน่วยงาน(Development of Personal Relation) เป็นต้น รายละเอียดของการวิ
เคราะห์การเปลี่ยนแปลงจะกล่าวเพิ่มเติมในหัวข้อ 2.1.3



รูปที่ 2.1 การพัฒนาระบบสารสนเทศ.

การวิเคราะห์และออกแบบระบบสารสนเทศ (Analysis and Design of
Information System) เป็นขั้นตอนที่มีจุดประสงค์ในการสร้างโมเดล (model) ซึ่งจะถูกใช้
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นชอบใช้เอกสารนี้
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นพื้นฐานในการสร้างระบบสารสนเทศต่อไป ขั้นตอนเหล่านี้สามารถแบ่งออกเป็น 4 ส่วนดังนี้

- 1 ศึกษากิจกรรม (Activity studies)
- 2 วิเคราะห์ข้อมูลเทคโนโลยี (Information analysis)
- 3 ออกแบบระบบข้อมูล (Data system design)
- 4 การเลือกใช้อุปกรณ์, เครื่องมือที่เหมาะสม (Equipment adaptation)

ส่วน 1 และ 2 เป็นขั้นตอนที่มุ่งในด้านกว้างที่จะบ่งชี้ว่าระบบสารสนเทศควรจะต้องทำอะไรหรือควรประกอบไปด้วยอะไรบ้างในการที่จะใช้แก้ปัญหา

ส่วน 3 และ 4 มุ่งไปในการเลือกอุปกรณ์ และกรรมวิธีในการจัดการข้อมูล (Data Processing) จะขออธิบายแต่ละส่วนโดยคร่าวๆดังนี้

1 ศึกษากิจกรรม

เป็นขั้นตอนในการจำกัดขอบเขตของระบบสารสนเทศที่เราจะสร้างขึ้นในอนาคต รวมไปถึงการให้ลำดับความสำคัญก่อนหลัง เพื่อที่จะให้ระบบสารสนเทศดังกล่าวสามารถแก้ปัญหาและตอบสนองความต้องการของกลุ่มบุคคลที่เกี่ยวข้องได้ คำว่า " จำกัดขอบเขต " หมายถึงว่าอินพุตเซต (Input Set) เอาท์พุตเซต (Output Set) และ คุณสมบัติที่เกี่ยวข้องกับเซตของข้อมูลจะต้องถูกกำหนดขึ้น การจำกัดขอบเขตนั้นจะไม่มีติดตามอุปกรณ์เทคโนโลยีที่มีอยู่แต่จะยึดตามปัญหาและความต้องการของผู้ใช้ โดยทำการศึกษากิจกรรมในระบบงานปัจจุบันอย่างละเอียดมากกว่าที่ทำในขั้นตอนวิเคราะห์การเปลี่ยนแปลง ทำการศึกษาระบบงานย่อยต่างๆว่ามีอะไรบ้าง และแต่ละระบบมีต้นทุนและผลตอบแทนเป็นอย่างไร

การศึกษากิจกรรมและระบบงานย่อยของงานปัจจุบัน จะเห็นความสำคัญที่ผู้ใช้ เนื่องจากผู้ใช้ระบบ คือผู้ที่รุ่งานดีที่สุดในว่าปัญหาเกิดขึ้นที่ใด ทั้งยังเป็นผู้ที่ต้องรับผิดชอบและทำงานในระบบใหม่ต่อไปอีกด้วย ถ้าหากผู้ใช้ไม่ให้ความร่วมมือในการพัฒนาระบบสารสนเทศแล้ว ก็ยากที่จะประสบผลสำเร็จได้

2 การวิเคราะห์ข้อมูลเทคโนโลยี

เป็นการให้การอธิบายหรือบ่งชี้ว่าระบบสารสนเทศที่ได้ทำการจำกัดขอบเขตในหัวข้อ 1 แล้วนั้น ควรประกอบด้วยอะไร และควรทำอะไรได้นบ้าง การบ่งชี้นี้จะถูกนำไปใช้ด้วยจุดประสงค์สองประการคือ ประการแรก ใช้เป็นเครื่องมืออำนวยความสะดวกในการสื่อสารระหว่างกลุ่มต่างๆที่มีส่วนเกี่ยวข้องกับระบบงานที่กำลังทำ และประการที่สองใช้เป็นพื้นฐานสำหรับการออกแบบระบบข้อมูล

ในการวิเคราะห์จะต้องกระทำกับงานย่อยของทุกระบบที่เกี่ยวข้อง ซึ่งการวิเคราะห์จะขยายขอบเขตไปมากเพียงใดก็ขึ้นอยู่กับว่า ระบบงานย่อยนั้นสามารถจัดระเบียบแบบแผนได้หรือไม่ และสามารถนำคอมพิวเตอร์เข้ามาช่วยได้หรือไม่ ถ้าระบบใดไม่สามารถทำได้ก็ให้ทำการวิเคราะห์เพียงคร่าวๆ เพียงเพื่อให้เข้าใจกระบวนการทำงานก็พอ ถ้าทำได้ก็จำเป็นที่จะต้องวิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เคราะห์ที่กันอย่างละเอียดถี่ถ้วน

3 การออกแบบระบบข้อมูล

เป็นการนำผลจากขั้นตอนที่แล้วมาใช้ในการออกแบบระบบข้อมูล โดยยังไม่มีการพิจารณาว่าจะใช้กับอุปกรณ์ใดๆ เป็นแต่เพียงการทำในลักษณะของหลักการว่าส่วนงานที่ทำได้โดยคนจะเป็นอย่างไร มีงานอะไรเกี่ยวข้องกับบ้าง ส่วนงานที่เป็นอัตโนมัติใช้คอมพิวเตอร์ช่วยเป็นอย่างไร โครงสร้างของข้อมูลและโปรแกรมเป็นเช่นไร โดยจะต้องมีการพิจารณาถึงการบำรุงรักษา ความคล่องตัว และการปฏิบัติงานได้อย่างถูกต้องด้วย

4 การเลือกใช้อุปกรณ์, เครื่องมือที่เหมาะสม

มีจุดมุ่งหมาย เพื่อที่จะเลือกเครื่องคอมพิวเตอร์หรืออุปกรณ์ที่จะนำมาใช้ให้เหมาะสมกับผลที่ได้ในข้อ 3 ดังนั้นส่วนนี้จึงเกี่ยวข้องกับส่วนงานอัตโนมัติของระบบสารสนเทศเท่านั้น

การสร้างระบบและการนำไปใช้งาน (Realization and Implementation of Information System) การสร้างระบบเป็นขั้นตอนที่จัดสร้างระบบสารสนเทศขึ้นตามโมเดลของระบบที่ได้มาจากการวิเคราะห์และออกแบบระบบสารสนเทศ โดยประกอบด้วยขั้นตอนสำคัญห้าขั้นตอน ดังนี้

- การเขียนหรือสร้างโปรแกรม
- การจัดทำแฟ้มข้อมูล
- การออกแบบงานที่ทำด้วยมือ
- การทดสอบระบบ
- การสร้างคู่มือ

เมื่อสร้างโปรแกรมและทดสอบระบบเสร็จแล้ว ก็มาถึงขั้นการใช้งานจริง ซึ่งในขั้นนี้จะต้องมีการเตรียมตัวในด้านต่าง ๆ อันได้แก่

- การจัดเตรียมบุคลากร ตลอดจนการฝึกอบรม
- การจัดพิมพ์แบบฟอร์มต่าง ๆ
- การจัดทำและเปลี่ยนแปลงแฟ้มข้อมูลเข้าสู่ระบบใหม่
- การใช้งานระบบ

แม้ว่าถึงตอนนี้ระบบใหม่จะทำงานได้ แต่ผู้ที่เกี่ยวข้องกับการสร้างระบบควรจะต้องติดตามผลลัพธ์ของระบบที่สร้าง เพื่อให้ได้ความรู้และประสบการณ์สำหรับเป็นแนวทางในการพัฒนาระบบอื่น ๆ ต่อไปให้ดียิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะเด่นหรือข้อดีของไอแอล

- 1 ระบบสารสนเทศจะถูกพัฒนาขึ้นก็ต่อเมื่อมีความจำเป็นจริง (รู้ได้จากขั้นตอนวิเคราะห์การเปลี่ยนแปลง)
- 2 ระบบสารสนเทศจะถูกพัฒนาดำเนินว่าคุ้มที่จะทำ คือให้ผลดีแก่กิจกรรมต่างๆในองค์กร (รู้ได้จากขั้นตอนการศึกษากิจกรรม)
- 3 ระบบสารสนเทศจะถูกพัฒนาขึ้นโดยให้ผู้ใช้มีส่วนร่วม คือผู้ใช้จะต้องรู้ว่าระบบสารสนเทศดังกล่าวประกอบด้วยอะไรและทำงานอย่างไร (ขั้นตอนการวิเคราะห์ข้อมูล)
- 4 ระบบสารสนเทศถูกพัฒนาขึ้นให้สามารถปรับปรุงเปลี่ยนแปลงได้เมื่อจำเป็น และจะไม่ขึ้นอยู่กับการใช้อุปกรณ์คอมพิวเตอร์แบบใด (ขั้นตอนการออกแบบระบบข้อมูล)
- 5 ระบบสารสนเทศถูกพัฒนาขึ้นมาโดยที่สามารถนำไปใช้กับอุปกรณ์เทคนิคได้อย่างมีประสิทธิภาพและเหมาะสม (ขั้นตอน การเลือกใช้อุปกรณ์, เครื่องมือที่เหมาะสม)
- 6 การพัฒนาระบบสารสนเทศจะถูกควบคุมโดยตัวของผู้ใช้เอง
- 7 การพัฒนาระบบสารสนเทศจะทำในลักษณะขั้นตอนย่อยๆ ซึ่งเป็นข้อดีในการแก้ปัญหาที่ยุ่งยากซับซ้อน
- 8 ทำให้เกิดความร่วมมือที่ดีระหว่างกลุ่มบุคคลต่างกลุ่มที่มีส่วนร่วมในการพัฒนาระบบสารสนเทศ ซึ่งทำให้การพัฒนาเกิดผลสำเร็จ
- 9 การปรับปรุงเปลี่ยนแปลงและบำรุงรักษาระบบสามารถกระทำได้อย่างรวดเร็วและถูกต้องปลอดภัย
- 10 การพัฒนาระบบสารสนเทศในที่สุดจะกลายเป็นกิจกรรมปกติของผู้ใช้ ผู้ใช้จะมีความคุ้นเคยกับระบบสารสนเทศ
- 11 นักวิเคราะห์ระบบจะเป็นผู้สนับสนุนวิธีการที่ใช้ในการพัฒนาระบบสารสนเทศ
- 12 ผู้ทำหน้าที่ออกแบบข้อมูลจะสามารถมุ่งความสนใจไปอยู่ที่งานของตนเองได้ ซึ่งจะเกิดผลดีในระยะยาว

2.1.2 แอคทิวิตี กราฟ (Activity Graph) หรือ เอ-กราฟ (A-Graph)

เอ-กราฟเป็นแผนภาพที่มีหน้าที่ในการอธิบายกิจกรรมต่างๆในหน่วยงาน โดยถูกใช้เป็นเครื่องมือสำคัญในขั้นตอนต่างในการพัฒนาระบบสารสนเทศตามวิธีการของไอแอล (ในขั้นตอนการวิเคราะห์การเปลี่ยนแปลงและขั้นตอนการศึกษากิจกรรม)

สัญลักษณ์ที่ใช้ใน เอ กราฟ ใช้แทนสิ่งต่างๆต่อไปนี้

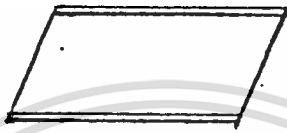
1 เซต (Set) ได้แก่เซตของสิ่งต่างๆ ทั้งบุคคล(Person) วัตถุสิ่งของ(Material) และข่าวสาร(Message) เซตที่เคลื่อนเข้าสู่กิจกรรม เราเรียกว่าอินพุต (Input) ส่วนเซตที่เคลื่อนออกจากกิจกรรมเราเรียกว่าเอาต์พุต (Output) ของกิจกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2 กิจกรรม (Activity) ทั้งที่กระทำโดยมนุษย์ , เครื่องมืออุปกรณ์ หรือประกอบกัน

3 โฟล (Flow) แลกเปลี่ยนไหลของ เขต ระหว่าง กิจกรรม ต่างๆ

สัญลักษณ์ที่ใช้แทนสิ่งทั้ง 3 ดังกล่าวร่วมกับข้อความต่างๆ ประกอบกันขึ้นเป็น เอ-กราฟ สัญลักษณ์ที่ใช้ใน เอ-กราฟ มีรูปแบบดังในรูป 2.2 และ 2.3



เซตที่แท้จริง (REAL SET)

: เขตของบุคคล และ สิ่งของ



เซตของข่าวสาร (MESSAGE SET)

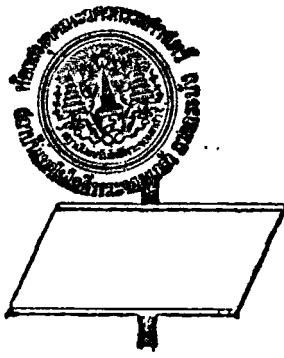
: เขตของข่าวสาร เช่น เอกสารต่างๆหรือข้อมูลข่าวสารทางโทรทัศน์



คอมโพสิทเซต (COMPOSITE SET)

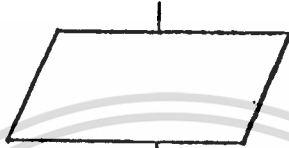
: เขต ที่เป็นทั้งเซตที่แท้จริง และเซตของข่าวสาร (เป็นบุคคลหรือวัตถุที่มาพร้อมข่าวสาร)

รูป 2.2 สัญลักษณ์ที่ใช้ในเอ-กราฟ



ไหลที่แท้จริง (Real Flow)

: การไหลของบุคคล หรือวัตถุสิ่งของ



ไหลของข่าวสาร (MESSAGE FLOW)

: การไหลของข่าวสาร



คอมโพสิทไฟล (COMPOSITE FLOW)

: การไหลของบุคคล หรือวัตถุสิ่งของ พร้อมข่าวสาร

กิจกรรม (ACTIVITY)

: กิจกรรมในองค์กรนั้นๆ

รูป 2.3 สัญลักษณ์ที่ใช้ในการเอ-กราฟ(ต่อ)

จุดเด่นที่สำคัญของเอ-กราฟ คือสามารถจำลองหรืออธิบายการทำงานของระบบได้ละเอียด
มากเท่าที่ผู้วิเคราะห์ต้องการเอ-กราฟที่ถูกเขียนขึ้นมาแรกสุด และมีความละเอียดน้อยที่สุดเรียก
ว่า แผนภูมิสรุป (Overview Graph) จากนั้นถ้าผู้วิเคราะห์ต้องการอธิบายส่วนย่อยต่างๆ ให้ชัดเจน
ขึ้นอีกก็สามารถจะกระทำได้โดยการเขียนเอ-กราฟ อธิบายหรือแสดงให้ละเอียดสักลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้จัดทำเอกสารนี้ได้รับแจ้งการคัด

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ร้อยละเป็นลำดับไป ซึ่งเอ-กราฟในลักษณะนี้ถูกเรียกว่าแผนภูมิรายละเอียด (Detail Graph) แผนภูมิรายละเอียดอาจจะถูกเขียนขึ้นหลายระดับ แล้วแต่ระดับความละเอียดในการใช้งาน ตัวอย่างของทั้งแผนภูมิลูกรูปและแผนภูมิรายละเอียดแสดงไว้ดังรูป 2.4.2.5

จากตัวอย่างแผนภูมิลูกรูปที่แสดงไว้ ได้เขียนขึ้นจากระบบงานของสหกรณ์ผู้ผลิตนมแห่งหนึ่งซึ่งรวมตัวกันขึ้นจากผู้ผลิตหลายราย สหกรณ์นี้มีศูนย์กลางอยู่ที่อพนพิศกลางอันเป็นสถานที่ประชุมวางแผนการดำเนินงานทางผู้ผลิตแต่ละรายจะรับผิดชอบการผลิตและการจำหน่ายเองการขายปลีกนั้นจะผ่านพ่อค้าคนกลาง ฝ่ายสหกรณ์จะมีการติดสื่อสารกับผู้บริหารในตำแหน่งชาวลากรเช่นการโฆษณา เป็นต้น ระบบที่เราสนใจ (ระบบที่เกิดปัญหาและต้องแก้ไข) ได้แก่ระบบการรับคำสั่งซื้อและการส่งสินค้าไปยังลูกค้า

จากเอ-กราฟในรูป อพนพิศกลาง (กิจกรรม 3) จะรับรู้เอาข่าวล้าร์เกี่ยวกับตลาดนมจากภายนอกระบบ (เซต 1A) ข่าวล้าร์นี้อาจได้จากสื่อมวลชน, ผู้ผลิตนมรายอื่นหรือจากแหล่งข่าวอื่น และยังได้รับรายงานทางด้านการเงินจากสมาชิกของสหกรณ์ (4A) ทางอพนพิศกลางจะใช้ข้อมูลทั้งสองเป็นเครื่องช่วยกำหนดแผนและแนวทางดำเนินงานให้กับสมาชิก (3B) และมีการแจ้งข่าวล้าร์ความเคลื่อนไหวของกิจกรรมต่างๆไปยังลูกค้า คือ ผู้บริโภคและพ่อค้าคนกลาง (3A) รวมทั้งข่าวล้าร์ประชาสัมพันธ์ไปยังภายนอกระบบ (3C) เช่น ให้ข่าวแก่หนังสือพิมพ์ธุรกิจ เป็นต้น ทางด้านผู้ผลิตนม (4) จะได้รับวัตถุดิบในการผลิต เช่น น้ำนมดิบ, ส่วนผสม (เช่น น้ำตาล) และบรรจุภัณฑ์ เป็นต้น (2A) ตลอดจนรับแผนงานและแนวทางดำเนินงานจากอพนพิศกลาง (3B) นอกจากนี้ ยังมี คำสั่งซื้อ, เงินค่าสินค้า และสินค้าส่งกลับคืนจากพ่อค้าด้วย (5B) และแน่นอนครับ สิ่งที่ได้ออกมาจากกิจกรรมนี้ก็คือ สินค้านมที่ผลิตแล้ว ส่งไปยังพ่อค้าพร้อมกับใบอินวอยซ์ (4B) และส่งรายงานด้านการเงินกลับไปยังอพนพิศกลาง (4A) ใส่งมายังพ่อค้าคนกลาง (5) ใรับสินค้ากลับในอินวอยซ์จากผู้ผลิตนม (4B) และรับข่าวล้าร์จากอพนพิศกลาง เช่น การจัดแคมเปญส่งเสริมการขาย เป็นต้น (3A) แล้วทางพ่อค้าจะส่งนมไปยังผู้บริโภค (5A) และส่งคำสั่งซื้อ, เงินชำระค่าสินค้า และสินค้ากลับคืนไปยังผู้ผลิต (5B) สุดท้าย ได้แก่ ผู้บริโภค (6) ใรับข่าวล้าร์ เช่น โฆษณาสินค้าโตเร็วกว่าที่แม่ส่งจากอพนพิศกลาง (3A) โดยที่จะมีนมที่ไปซื้อมาไว้บริโภค (6A)

จากแผนภูมิลูกรูปของระบบรับคำสั่งซื้อและส่งสินค้า... ต้องการขยายรายละเอียดของกิจกรรมหมายเลข 4 "ผู้ผลิตนม" ก็จะสามารถเขียนแผนภูมิรายละเอียดได้ดังรูป กรอบสี่เหลี่ยมใหญ่ก็เป็นเช่นเดียวกับแผนภูมิลูกรูป คือเป็นการแสดงขอบเขตของระบบที่สนใจ ซึ่งในกรณีนี้ขอบเขตดังกล่าวคือกิจกรรมหมายเลข 4 "ผู้ผลิตนม" ของแผนภูมิลูกรูป ภายในกรอบแบ่งเป็นกิจกรรมของแผนกต่างงั้นหมายความว่ามีการแบ่งแผนกของผู้ผลิตนม ซึ่งที่จริงจะแบ่งกิจกรรมเป็นอย่างไรก็ได้ เช่น อาจรวมหลายแผนกเป็นกิจกรรมเดียวกัน หรือแผนกเดียวแต่แยกเป็นหลายกิจกรรม ทั้งนี้ขึ้นอยู่กับพิจารณาตามความเหมาะสมในการใช้งานของผู้วิเคราะห์ นอกจากนี้ดังได้กล่าวไว้แล้วว่าแผนภูมิรายละเอียดมีได้หลายระดับ ดังนั้นผู้วิเคราะห์ก็ยังสามารถขยายรายละเอียดเพิ่มได้อีก เช่น นำกิจกรรม

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่ควรนำออกเผยแพร่โดยไม่ได้รับอนุญาต

ของแผนกเก็บรักษาและส่งคืนค่าไปเขียนเป็นเอ-กราฟภาพใหม่ ซึ่งจะเป็นแผนภูมิรายละเอียดระดับที่ 2 เป็นต้น แต่ในที่นี้จะขอแสดงไว้เพียงแค่ระดับที่ 1

ข้อกำหนดและรายละเอียดต่างๆในการเขียนเอ-กราฟมีดังนี้

1. ส่วนบนสุดของกระดาษที่เขียนเอ-กราฟ จะมีการระบุสิ่งต่างๆเป็นการให้รายละเอียดของเอ-กราฟที่เขียน โดยแบ่งออกเป็น 3 คอลัมน์อันได้แก่

คอลัมน์ที่ 1 ทางซ้ายมือสุด แบ่งได้เป็น 3 ส่วนย่อยคือ

- ส่วนที่ 1 นอกชื่อของหน่วยงาน,องค์กร ในที่นี้ สมมติเป็นสผน. (ซึ่งย่อมาจาก สหกรณ์ผู้ผลิตนม)

- ส่วนที่ 2 นอกชื่อบุคคล (หรือกลุ่มบุคคล) ผู้วิเคราะห์ระบบ

- ส่วนที่ 3 นอกขอบเขตระบบที่เอ-กราฟแสดง

คอลัมน์ที่ 2 ถัดมาบริเวณกลางหน้ากระดาษ แบ่งเป็น 2 ส่วนย่อยคือ

- ส่วนที่ 1 นอกให้ทราบว่าเป็นเอกสารนี้ คือ กราฟอะไร ซึ่งในที่นี้คือเอ-กราฟ(ตามหลักวิธีการของไอแอล จะมีการนำประเภทอื่นนอกเหนือไปจากเอ-กราฟ แต่เพราะเหตุผลที่ไม่ได้นำมาใช้ในโครงการ จึงไม่ขอกล่าวถึงในบริบทนี้) /

- ส่วนที่ 2 นอกวันที่ที่เขียนหรือสร้างเอ-กราฟ (ถ้าเป็นเวอร์ชันที่ปรับปรุงก็เป็นวันที่ทำการปรับปรุง)

คอลัมน์ที่ 3 ทางขวามือสุด แยกเป็น 2 ส่วนย่อย

- ส่วนที่ 1 นอกเลขอ้างอิงของเอ-กราฟ โดยมุ่งถึงกิจกรรมที่เอ-กราฟนั้นอธิบาย

- ส่วนที่ 2 นอกเวอร์ชันที่ปรับปรุง(ทุกครั้งที่ทำกรปรับปรุงควรมีการให้เลขเวอร์ชันใหม่)

2. กรอบสี่เหลี่ยมที่ล้อมรอบกิจกรรมทั้งหลายอยู่ภายใน (Graph Frame) จะทำหน้าที่แสดงขอบเขตของระบบงานที่เราศึกษา กิจกรรมทั้งหลายดังกล่าวจึงเป็นกิจกรรมภายในขอบเขตของระบบงาน

3. สำหรับขีดต่างๆจำแนกโดยอาศัยตำแหน่งที่อยู่เปรียบเทียบกับกรอบสี่เหลี่ยมจะมีอยู่ 3 ลักษณะคือ หากแรกจะอยู่ด้านบนนอกกรอบสี่เหลี่ยม ซึ่งในกรณีเอ-กราฟดังกล่าวจะเป็น แผนภาพสรุปเขตดังกล่าวจะเป็นอินพุตที่เกิดจากกิจกรรมภายนอกระบบ แต่มาเป็นอินพุตของกิจกรรมภายในระบบที่กำลังทำการศึกษา แต่ถ้าเป็นแผนภูมิรายละเอียด เขตเหล่านี้จะเป็นเขตที่ถูกอ้างอิงถึงมาแล้วจากแผนภูมิสรุป หรือแผนภูมิรายละเอียดที่อยู่เหนือขึ้นไป ตัวอย่างสำหรับแผนภูมิสรุป C0 ได้แก่เขต1A และเขต2A ส่วนตัวอย่างสำหรับแผนภูมิรายละเอียด C4 ได้แก่ เขต3B1 เขต3B2 เขต3B3

ส่วนต่อมาก็คือหากที่เป็นเอาต์พุตที่เกิดจากกิจกรรมภายในระบบ แต่ยังไม่หลุดออกนอกระบบ โดยเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ดูแลระบบใช้เอกสารนี้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้เข้าไปเป็นอินพุตของกิจกรรมชั้นภายในระบบอีกที พวกนี้จะอยู่ภายในกรอบสี่เหลี่ยม ดังเช่น
เซต 3A เซต 5B เป็นต้น สำหรับพวกลูกข่ายเป็นเซตที่เกิดจากกิจกรรมในระบบเช่นเดียวกับ
พวกที่สอง แต่หันจากระบบไปเป็นอินพุตของกิจกรรมในระบบอื่นภายนอก เซตพวกนี้ จะอยู่ด้าน
ล่างนอกกรอบสี่เหลี่ยม ตามตัวอย่าง ได้แก่ เซต 3C และเซต 6A

4. เลขที่อ้างอิงของเอ-กราฟจะประกอบด้วยตัวอักษรย่อหน้าหน้า โดยตัวอักษรย่อหน้าจะใช้อ้างอิง
เอ-กราฟทุกภาพในระบบ โดยอาจจะตั้งขึ้นด้วยความหมายใดก็ได้ เช่นชื่อย่อกลุ่มผู้วิเคราะห์ระบบ
เป็นต้น ในการนี้ตัวอย่างนี้ใช้เป็นตัว C และมีตัวเลขกำกับท้าย สำหรับแผนภูมิสรุปซึ่งในระบบนี้
เพียงภาพเดียวจะใช้เลข ๑ เสมอ ส่วนแผนภูมิละเอียดจะใช้เลขอ้างอิงของกิจกรรมที่นำ
มาเขียนเป็นแผนภูมิละเอียดนั้นเป็นเลขกำกับท้าย จากตัวอย่างเลขที่กำกับท้ายแผนภูมิละเอียด
ในรูป 2.5 ก็คือ C4 เพราะนำมาจากกิจกรรมหมายเลข 4 ในแผนภูมิสรุปในรูป 2.4 โดยจะนำ
เลขอ้างอิงของเอ-กราฟ มาเขียนกำกับไว้ ที่ขอบซ้ายบนของกรอบสี่เหลี่ยม

5. สำหรับเซตต่างๆที่มีตัวเลขและตัวอักษรใช้อ้างอิงเช่นเดียวกัน โดยมีหลักเกณฑ์เบื้องต้นคือดูว่า
เซตนั้นเป็นเอาต์พุตของกิจกรรมใด ก็ใช้เลขอ้างอิงของกิจกรรมนั้นประกอบกับตัวอักษรภาษา
อังกฤษ A, B, ... เป็นตัวอ้างอิงเซต ตัวอย่างเช่น กิจกรรมหมายเลข 3 มีเอาต์พุต 3 เซต
ก็จะได้ 3A, 3B และ 3C เป็นตัวอ้างอิงเป็นต้น โดยมีหลักเพิ่มเติมดังนี้คือ สำหรับแผนภูมิละเอียด
ละเอียดนั้น เซตที่อยู่ภายนอกกรอบสี่เหลี่ยมไม่ว่าด้านบนหรือด้านล่างก็ตาม (อินพุตเซต หรือ
เอาต์พุตเซตของแผนภาพ) ส่วนได้แก่บรรดาเซตหรือเซตย่อยที่ถูกอ้างอิงมาแล้วจากเอ-กราฟที่อยู่
เหนือแผนภูมิละเอียดนี้ให้ใช้เลขอ้างอิงคงเดิม ในกรณีแผนภาพสรุปการให้เลขอ้างอิงจะมีลักษณะ
พิเศษอยู่คือเนื่องจากเป็นแผนภาพที่เขียนขึ้นมาโดยเริ่มแรก อินพุตที่เข้าสู่แผนภาพจึงไม่ถูกอ้างอิงที่
ใดมาก่อน ดังนั้นการให้เลขอ้างอิงจะให้โดยถือเสมือนว่าเซตเหล่านี้เกิดจากกิจกรรมลมมุตติที่อยู่ใน
แผนภาพสรุปนั่นเอง ดังในภาพ 2.4 เซต 1A และ 2A ถูกอ้างอิงโดยถือว่าเกิดจากกิจกรรม
ลมมุตติ 1 และ 2 ภายในแผนภาพสรุป

6. แต่ละกิจกรรมภายในระบบจะมีหมายเลขเพื่อใช้อ้างอิงที่แตกต่างกันและมีชื่อนอกไว้ด้วยเช่น
กิจกรรมหมายเลข 3 คือ กิจกรรมในออฟฟิศกลาง กิจกรรมหมายเลข 6 คือ กิจกรรมของผู้บริ
โภคเป็นต้น การกำหนดเลขอ้างอิงของกิจกรรมให้กำหนดเลขเรียงลำดับกันจากน้อยไปมาก โดย
ใส่เรียงกิจกรรมที่มีตำแหน่งบนเอ-กราฟจากบนลงล่าง และจากซ้ายไปขวา สำหรับแผนภาพสรุป
จะถือว่ามีการให้อ้างหมายเลขอ้างอิงแก่กิจกรรมลมมุตติที่กล่าวไว้ในหัวข้อที่ 5 ด้วยและการให้
หมายเลขอ้างอิงแก่กิจกรรมที่มีอยู่จริงก็จะให้ต่อท้ายกิจกรรมลมมุตติดังกล่าว ดังในภาพ 2.4
เลขอ้างอิงของกิจกรรมเริ่มที่ 3 ต่อท้ายกิจกรรมลมมุตติ 1 และ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. สำหรับกิจกรรมย่อย หรือ เขตย่อย จะมีการใช้เลขอ้างอิงเต็มที่ใช้กับกิจกรรม หรือเขตที่มันแบ่งย่อยออกมา โดยมีการเพิ่มตัวเลขขึ้นอีกหนึ่งหลัก แต่ในการแสดงเขต หรือกิจกรรมย่อยเหล่านี้ในเอ-กราฟ จะมีการใช้ " - " นำหน้าตัวเลขที่เพิ่มหลักขึ้นมาเป็นและหมายเลขอ้างอิงที่นำหน้าเอาไว้

8. จำนวนกิจกรรมย่อยในแต่ละเอ-กราฟจะต้องไม่เกินเก้ากิจกรรม มิฉะนั้นถ้าหากมีการอ้างอิงถึงกิจกรรมที่สับสนควรใช้ตัวเลขอ้างอิงอาจทำให้เกิดความสับสน การนับจำนวนไม่เกินเก้านี้ สำหรับแผนภาพสรุปจะนับรวมกิจกรรมสมมุติด้วย

9. ห้ามเส้นแสดงการเคลื่อนที่ (flow) ของเซตเชื่อมต่อกันระหว่างกิจกรรมหนึ่งไปยังกิจกรรมหนึ่ง หรือเซตหนึ่งไปยังเซตหนึ่งโดยตรง ต้องเชื่อมจากกิจกรรมไปยังเซต หรือจากเซตไปยังกิจกรรมเท่านั้น

10. จำนวนอินพุตเซตของกิจกรรมแต่ละกิจกรรมจะต้องไม่เกินหก และจำนวนเอาต์พุตเซตของกิจกรรมแต่ละกิจกรรมจะต้องไม่เกินหก ลาเหตุที่ต้องจำกัดจำนวนนั้นเท่าที่ได้ทดลองใช้ในโครงการเข้าใจว่าเมื่อไม่ให้เอ-กราฟดูสับสนยุ่งเหยิง ดูได้ยากประการหนึ่ง อีกประการหนึ่งคือเพื่อให้ผู้เขียนเอ-กราฟพยายามรวมกิจกรรมและเซตต่างๆ เข้าด้วยกัน การจะแจกแจงรายละเอียดส่วนใดลงไปก็ให้ทำการเขียนแผนภูมิลายละเอียดในส่วนนั้น มิใช่เขียนแสดงโดยละเอียดในที่เดียว ซึ่งจะเป็นการทำลายข้อดีเด่นของเอกราฟไป

11. แผนภูมิลายละเอียดต้องมีอินพุตเซต เอาต์พุตเหมือนเอ-กราฟที่อยู่เหนือมันขึ้นไป (เหมือนแต่จะละเอียดกว่า)

รายงานกำกับ (TEXT PAGE)

ในการใช้งานจริงๆแล้ว เอ-กราฟยังมีอุปสรรคและข้อจำกัดอยู่ เพราะหน้ากระดาษมีเนื้อที่จำกัด ผู้เขียนเอ-กราฟจะถูกบังคับทางอ้อม 2 ประการ ได้แก่ ประการแรก การเขียนข้อความโดยต้องให้สั้นและกะทัดรัดเข้าใจ อย่างเช่นกิจกรรมหมายเลข 5 ต้องใช้คำว่า "ผู้ผลิตนม" แทนข้อความ "การผลิตและจำหน่ายสินค้าของผู้ผลิตนม" เป็นต้น ส่วนประการที่สองก็คือ หากสิ่งที่เราสนใจ (เซต) มีมาก จะต้องพยายามรวมสิ่งเหล่านั้นเข้าไว้เป็นเซตเดียวกัน เพื่อให้มีจำนวนเซตน้อยที่สุด เช่น เซต 5B แทนที่จะแยกเป็นเซตของคำสั่งซื้อ เซตของเงินค่าสินค้า และเซตของสินค้าส่งคืน ก็ต้องนำมารวมเป็นเซตเดียว ข้อจำกัดดังกล่าว ส่งผลให้เอ-กราฟอาจรวบรัดเกินไป จนผู้ดูทำความเข้าใจได้ยาก หรือไม่ก็ต้องทำการดูแผนภูมิลายละเอียดเพิ่มเติม (ซึ่งก็อาจจะยังไม่ได้เขียน) ไอแซ็กจึงได้ตกลงกำหนดให้มีการทำเอกสารเสริมขึ้นมาชุดหนึ่ง เรียกว่า ทราย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่ควรนำออกเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานกำกับซึ่งใช้สำหรับแจกแจงรายละเอียดของเขตและของกิจกรรมในเอ-กราฟว่า มีเขตและกิจกรรมอะไรบ้าง โดยมีข้อความอธิบายเพิ่มเติมจากข้อความที่เขียนไว้สั้นๆในเอ-กราฟ และถ้าเขตใดมีส่วนประกอบที่ต้องการแจกแจง ก็อาจเขียนแยกเป็นเขตย่อยไว้ให้ ตัวอย่างรายงานกำกับได้แสดงไว้ในหน้าถัดไปซึ่งเป็นรายงานกำกับของ เอ-กราฟ C0 และ C4 ตามลำดับ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สผน.

รายงานเกี่ยวกับ

ผู้วิเคราะห์ :

เอ-กราฟ

วิทยา ต่อศรีเจริญ

เลขอ้างอิง C๒

ขอบเขต :

วันที่ : 13 พย. 2532

ระบบรับคำสั่งซื้อและส่งสินค้า

เวอร์ชัน 4

1A ข่าวสารเกี่ยวกับตลาดนม

1A1 ข่าวสารทางสื่อมวลชน

1A2 ข่าวสารจากแหล่งอื่น

2A วัตถุประสงค์

2A1 น้ำนมดิบ

2A2 ส่วนผสม เช่น น้ำตาล เป็นต้น

2A3 นวัตกรรม

3 การวางแผนและแนวทางการดำเนินงาน ณ ออฟฟิศกลาง

3A ข่าวสารที่ส่งให้กับพ่อค้าคนกลางและผู้บริโภค

3A1 โฆษณาสำหรับพ่อค้าคนกลาง (ไปยังกิจกรรมหมายเลข 5)

3A2 ข่าวเรื่องการจัดแคมเปญส่งเสริมการขาย (ไปยังกิจกรรมหมายเลข 5)

3A3 โฆษณาสำหรับผู้บริโภค เช่น ไดเรกต์เมลล์ (ไปยังกิจกรรมหมายเลข 6)

3B แผนงานและแนวทางการดำเนินงาน

3B1 แผนงาน

3B2 แนวทางการดำเนินงาน

4 การผลิตและจำหน่ายนมของผู้ผลิตนม

4A รายงานทางการเงินของผู้ผลิตนม

4B สินค้าที่ผลิตแล้วและในอินวอยซ์ที่จะส่งไปให้พ่อค้าคนกลาง

4B1 นมสำเร็จรูป

4B2 ในอินวอยซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 การรับและเก็บสินค้า ณ ที่ประกอบการของพ่อค้าคนกลาง

5A สินค้า (นม) ที่เก็บเอาไว้ในห้องเย็น รอให้ผู้บริโภคซื้อไป

5B คำสั่งซื้อ, เงินค่าสินค้า และสินค้าที่ถูกส่งกลับคืน

5B1 คำสั่งซื้อ อาจเป็นใบสั่งซื้อหรือสั่งทางโทรศัพท์

5B2 เงินชำระค่าสินค้า หลังจากที่อยู่พ่อค้าคนกลางได้รับสินค้าแล้ว

5B3 สินค้าที่เงื่อนไขไม่ตรงตามที่สั่งถูกส่งคืน

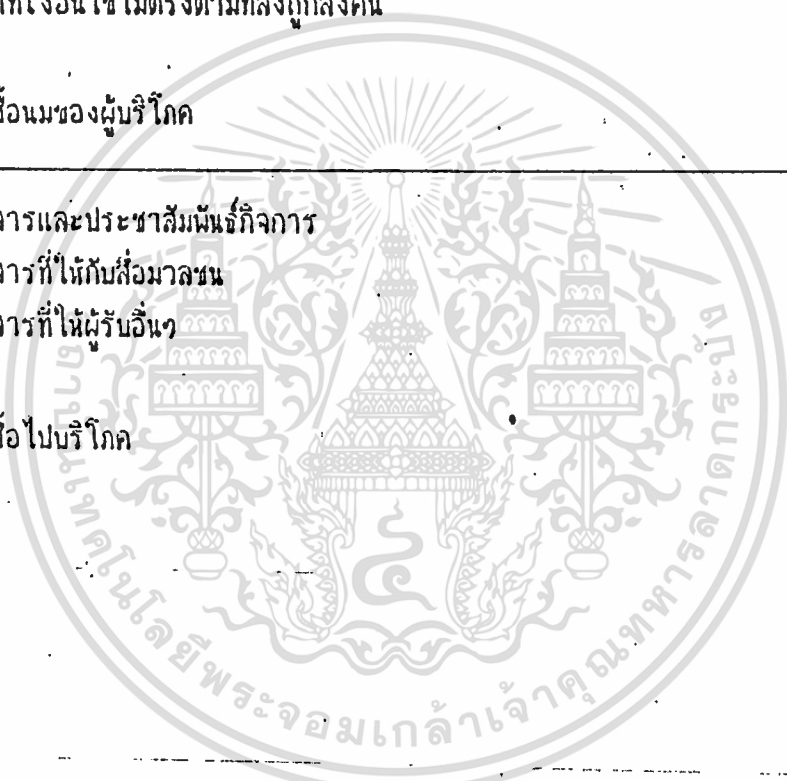
6 การซื้อนมของผู้บริโภค

3C ข้าราชการและประชาสัมพันธ์กิจการ

3C1 ข้าราชการที่ให้กับสื่อมวลชน

3C2 ข้าราชการที่ให้ผู้อื่นอื่น

6A นมที่ซื้อไปบริโภค



ลพณ.

รายงานกำกับ

ผู้วิเคราะห์:

เอ-กราฟ

วิทยา ต่อศรีเจริญ

เลขอ้างอิง C4

ขอบเขต :

วันที่ 9 ธค. 2532

ระบบรับคำสั่งซื้อและการส่งสินค้า : ผู้ผลิตนม

เวอร์ชัน 4

2A วัตถุประสงค์

2A1 น้ำนมดิบ

2A2 ส่วนผสม เช่น น้ำตาล เป็นต้น

2A3 นวัตกรรม

3B1 แผนงาน

3B11 แผนการผลิตระยะยาว

3B12 แผนควบคุมการผลิตระยะสั้น

3B2 แนวทางการดำเนินงาน

3B21 แนวทางการจัดระบบบัญชีและการออกใบอินวอยซ์ (ไปยังกิจกรรมหมายเลข 43)

3B22 แนวทางดำเนินงานสำหรับการเก็บรักษาและการส่งสินค้า (ไปยังกิจกรรมหมายเลข 44)

5B1 คำสั่งซื้อ อาจเป็นใบสั่งซื้อหรือส่งทางโทรศัพท์

5B11 ใบสั่งซื้อจากลูกค้าโดยตรง

5B12 คำสั่งซื้อทางโทรศัพท์

5B2 เงินชำระค่าสินค้า หลังจากที่มีลูกค้าตกลงได้รับสินค้าแล้ว

5B21 เงินชำระค่าสินค้า

5B22 ใบแสดงหลักฐานการชำระเงิน

5B3 สินค้าที่เงื่อนไขไม่ตรงตามที่สั่ง ถูกส่งคืนกลับมาพร้อมเอกสาร

5B31 ตัวสินค้าที่ถูกส่งคืนกลับมา

5B32 ใบแสดงหลักฐานการส่งคืน

4 ผู้ผลิตนม (การผลิตและจำหน่ายนมของผู้ผลิตนม)

- 1 แผนกรับคำสั่งซื้อจากลูกค้า

- 1A ข้อมูลคำสั่งซื้อสำหรับเขียนลงในใบอินวอยซ์

- 1B ข้อมูลคำสั่งซื้อสำหรับการผลิตและการส่งสินค้า เช่น จำนวนสินค้าที่สั่ง สถานที่ส่งสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

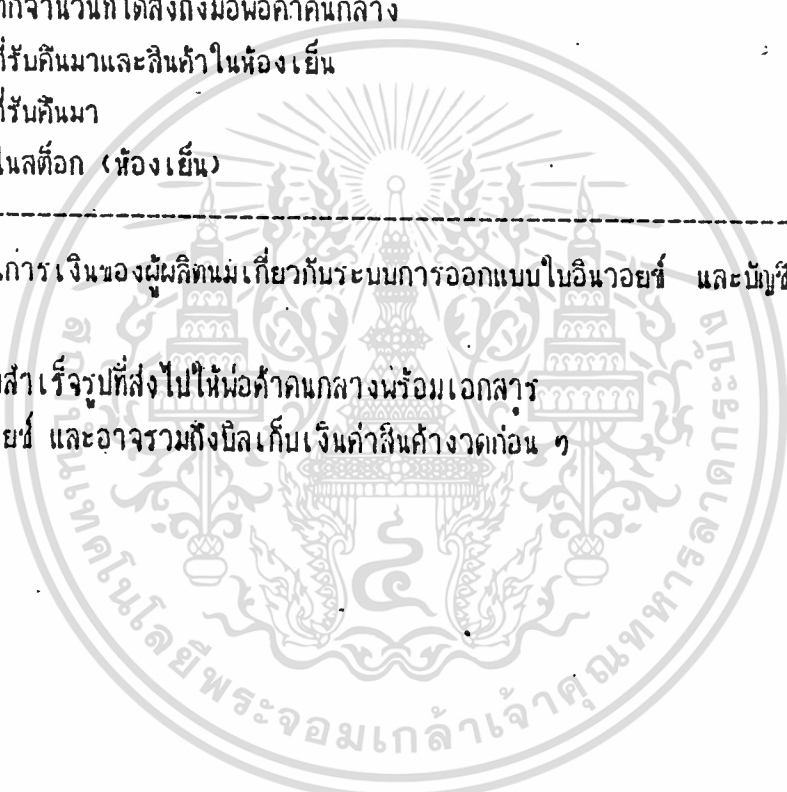
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2 แผนกผลิต
- 2A สินค้าใหม่ที่ผ่านการบรรจุเรียบร้อยแล้ว
- 3 แผนกบัญชีและใบอินวอยซ์
- 4 แผนกเก็บรักษาและส่งสินค้า
- 4A ข้อมูลจำนวนสินค้าในวาระ และสถานะต่าง ๆ คือ สินค้าที่ถูกสั่งซื้อ, ได้รับคืน และได้ส่งถึงมือค้า
- 4A1 ใบบันทึกจำนวนสั่งซื้อ
- 4A2 ใบบันทึกจำนวนรับคืน
- 4A3 ใบบันทึกจำนวนที่ได้ส่งถึงมือพ่อค้าคนกลาง
- 4B สินค้าที่รับคืนมาและสินค้าในห้องเย็น
- 4B1 สินค้าที่รับคืนมา
- 4B2 สินค้าในสต็อก (ห้องเย็น)

4A รายงานการเงินของผู้ถือหุ้นเกี่ยวกับระบอบการออกแบบใบอินวอยซ์ และบัญชีแยกประเภทต่าง ๆ

4B1 สินค้านมสำเร็จรูปที่ส่งไปให้พ่อค้าคนกลางพร้อมเอกสาร

4B2 ใบอินวอยซ์ และอาจรวมถึงบิลเก็บเงินค่าสินค้างวดก่อน ๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อกำหนดในการเขียนรายงานกำกับ มีเป็นข้อๆ ดังนี้

1. รายงานกำกับมีหัวกระดาษเช่นเดียวกับเอ-กราฟ แต่เพิ่มคำว่าเท็กซ์เพจตรงกลางหัวกระดาษ
2. ตัวเนื้อหาของรายงานกำกับแบ่งเป็น 3 ส่วน แต่ละส่วนแยกให้เห็นชัดจากกันด้วยเส้นตรงแนวนอน โดยที่
 - ส่วนที่ 1 อธิบายรายละเอียดของสิ่งที่อยู่นอกกรอบสี่เหลี่ยมด้านบนของเอ-กราฟ ตัวอย่างได้แก่ เซ็ต 1A และเซ็ต 2A
 - ส่วนที่ 2 อธิบายรายละเอียดของสิ่งที่อยู่ภายในกรอบสี่เหลี่ยมของเอ-กราฟ ตัวอย่างได้แก่กิจกรรมหมายเลข 3, 4, 5 และ 6 ตลอดจนเซ็ต 3A, 3B, 4A, 4B, 5A และ 5B
 - ส่วนที่ 3 อธิบายรายละเอียดของสิ่งที่อยู่ภายนอกกรอบสี่เหลี่ยมด้านล่างของเอ-กราฟ ตัวอย่างได้แก่ เซ็ต 3C และเซ็ต 6A
3. แต่ละเซ็ตภายในเอ-กราฟสามารถแบ่งเป็น เซ็ตย่อยเพื่อเจาะลึกถึงรายละเอียดของเซ็ตนั้นได้ (โดยที่เซ็ตย่อยเหล่านั้นไม่ได้แสดงไว้ในเอ-กราฟ) ตัวอย่าง ตามรูปที่ 2.6 เซ็ต 1A แบ่งได้เป็นเซ็ต 1A1 และเซ็ต 1A2 ส่วนเซ็ต 2A ก็แบ่งเป็นเซ็ต 2A1, 2A2 และ 2A3 แต่ไม่แสดงไว้ มาแจกแจงไว้ที่รูปที่ 2 การแบ่งเซ็ตย่อยจะแบ่งเท่าไรก็ขึ้นอยู่กับดุลพินิจของผู้เขียนเอ-กราฟเอง บางเซ็ตอย่างเช่นเซ็ต 4A และเซ็ต 5A อาจไม่ห้องแบ่งก็ได้
4. ในการเขียนจะเรียงลำดับหมายเลขอ้างอิงของเซ็ตและของกิจกรรมจากน้อยไปมากภายในแต่ละส่วนที่กล่าวไว้ในหัวข้อ 3 ในรายงานกำกับของ C4 นั้นส่วนที่ 1 เขียน 1A ก่อน 2A ส่วนที่ 3 ก็เขียน 3C ก่อน 6A สำหรับส่วนที่ 2 จะเขียนกิจกรรมที่มีเลขอ้างอิงน้อยก่อน ตามด้วยเซ็ตที่เป็นเอาต์พุตของกิจกรรมนั้น แล้วจึงเขียนกิจกรรมที่มีหมายเลขอ้างอิงถัดไปเรื่อยๆ ดังในตัวอย่าง
5. จะสังเกตเห็นว่าเวลาเขียนถึงเซ็ตและกิจกรรมที่มีอ้างอิงไว้ในเอ-กราฟ จะมีการเว้นบรรทัดเสมอ เฉพาะเซ็ตย่อยซึ่งเรามีอ้างอิงไว้ในเอ-กราฟจึงจะเขียนอยู่ที่เซ็ตใหญ่ของมัน โดยไม่มีการเว้นบรรทัด

จะเห็นว่ารายงานกำกับมีประโยชน์ต่อ เอ-กราฟอย่างมากในแง่ของการเป็นตัวเสริมให้เอ-กราฟมีประสิทธิภาพในการสื่อความหมาย ให้ผู้ดูสามารถเข้าใจระบบได้กระจ่างขึ้น

ตารางแสดงคุณสมบัติ (Property Tables)

จากเอ-กราฟและรายงานกำกับทำให้สามารถแสดงขั้นตอน, ลักษณะการดำเนินงาน แต่ไม่สามารถแสดงจำนวน, ปริมาณหรือขนาดของกรกฎเคลื่อนที่, จำนวนเอกสารที่เข้าและออกในแต่ละกิจกรรมได้ ซึ่งรายละเอียดเหล่านี้สามารถแสดงได้โดยตารางแสดงคุณสมบัติ

2.1.3 การวิเคราะห์การเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ในการทำงานตามโครงการนั้น ได้นำเพียงขั้นตอนการวิเคราะห์การเปลี่ยนแปลงของไอ แลคมมาใช้เท่านั้น ดังนั้นจึงขอกล่าวรายละเอียดเพิ่มเติมจากที่ได้กล่าวถึงขั้นตอนนี้แต่เพียงโดยย่อ ในหัวข้อ 2.1.1

การเปลี่ยนแปลงในระบบงานเพื่อปรับปรุงการทำงาน หรือแก้ปัญหาในองค์กรต่าง ๆ นั้น จะมีการเปลี่ยนแปลงได้ในหลายรูปแบบ หลายแนวทางหรือวิธีการ หากให้ผู้เชี่ยวชาญในสาขาต่าง ๆ กันมาเลือกแนวทางในการเปลี่ยนแปลง เขาเหล่านั้นก็ย่อมเสนอการเปลี่ยนแปลงในแนวทางที่ตนถนัด จุดประสงค์ของการวิเคราะห์การเปลี่ยนแปลงก็เพื่อรับแนวทางการเปลี่ยนแปลงเหล่านั้นเข้าด้วยกันอย่างเหมาะสม รวมไปถึงเป็นการหาต้นตอที่เกิดของปัญหาว่าอยู่ที่ไหนก่อนที่จะลงมือแก้ และสุดท้ายเป็นการหาคำตอบว่าแนวทางการเปลี่ยนแปลงต่างๆ ที่เสนอมานั้นจะให้ผลลัพธ์ตามที่ต้องการได้หรือไม่มากนักเพียงใด

หลักการสำคัญในการวิเคราะห์การเปลี่ยนแปลงมีดังนี้

- บุคคลจากกลุ่มบุคคลต่างที่เกี่ยวข้อกับตัวปัญหาจะต้องมีส่วนร่วมในการวิเคราะห์การเปลี่ยนแปลง เพราะเนื่องจากบุคคลเหล่านี้มีประสบการณ์แตกต่างกัน มองปัญหาในแง่มุมมองที่แตกต่างกัน
- ผลจากการวิเคราะห์การเปลี่ยนแปลงควรจะสามารถนำมาใช้ได้โดยตรงในงานชิ้นต่อไป
- ความเป็นไปได้ที่แนวทางการเปลี่ยนแปลงหนึ่งๆ จะสามารถทำงานได้หรือจัดสร้างขึ้นได้จะต้องถูกศึกษาในระหว่างการวิเคราะห์การเปลี่ยนแปลง
- ในการวิเคราะห์การเปลี่ยนแปลงจะมุ่งไปที่การแก้ปัญหา ข้อกำหนดกฎเกณฑ์ต่างๆ เป็นเพียงเครื่องมือช่วยในการแก้ปัญหาเท่านั้น อย่าให้สิ่งเหล่านี้มาเป็นอุปสรรคในการทำงาน

ขั้นตอนต่างๆในการวิเคราะห์การเปลี่ยนแปลง

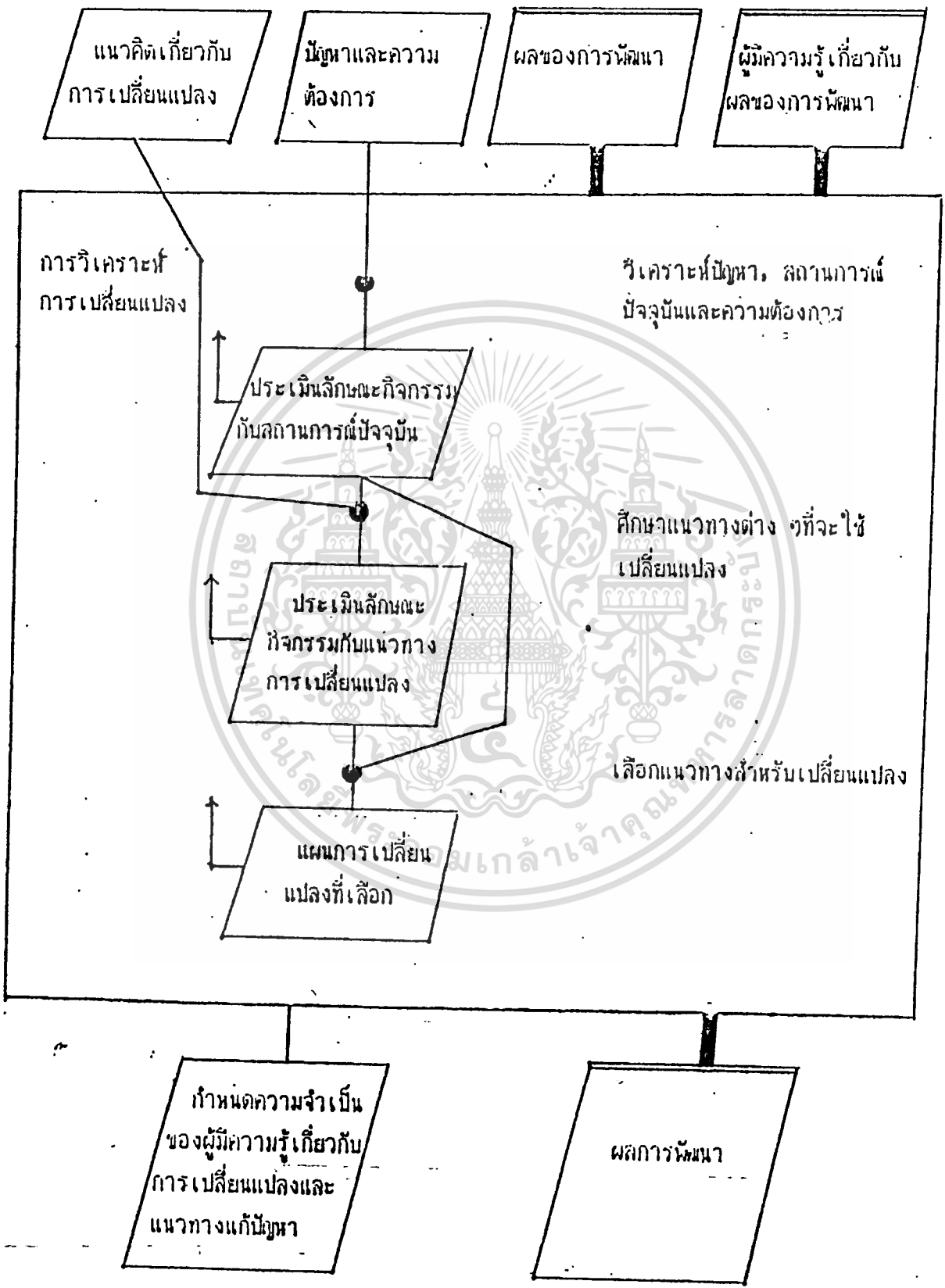
การวิเคราะห์การเปลี่ยนแปลงมีขั้นตอนอยู่ด้วยกันสามขั้นตอน เรียงตามลำดับการปฏิบัติดังนี้ คือ

- วิเคราะห์ปัญหา, สถานการณ์ปัจจุบันและความต้องการ (Analysis of Problems, Current Situation, and Needs)
- ศึกษาแนวทางต่าง ๆ ในการเปลี่ยนแปลง (Study of Change Alternatives)
- เลือกแนวทางในการเปลี่ยนแปลง (Choice of Change Approach)

ในรูป 2.6 ได้แสดงให้เห็นความสัมพันธ์ของขั้นตอนเหล่านี้และสิ่งต่างๆที่เกี่ยวข้อง

การวิเคราะห์ปัญหา, สถานการณ์ปัจจุบันและความต้องการ
งานในขั้นนี้แบ่งออกเป็นงานย่อยหกงานดังนี้ คือ

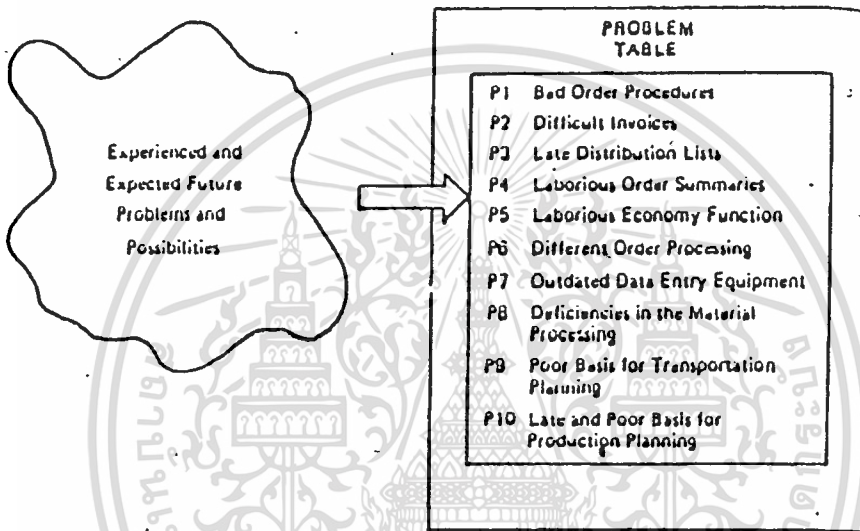
1. แสดงปัญหาที่มี (Problem listing)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น. ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น รูปที่ 2.6 วิธีการวิเคราะห์การเปลี่ยนแปลง เจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการค้นหาตรวจสอบปัญหาต่างๆที่เกิดขึ้นและที่มิโอกาสจะเกิดในอนาคต เพื่อแสดงให้เห็นว่าการเปลี่ยนแปลงที่ใช้มีประโยชน์ในการแก้ปัญหาในองค์กรอย่างไร โดยแสดงปัญหาต่างๆไว้ในตารางปัญหา (Problem Table) ซึ่งตารางนี้มีรูปแบบอย่างดังต่อไปนี้

- หมายเลขอ้างอิงปัญหา ที่ขึ้นต้นด้วยอักษร 'P' และตามด้วยตัวเลข
- ปัญหา
- คำอธิบายปัญหา



รูปที่ 2.7 การแสดงปัญหาที่มี

2. การวิเคราะห์กลุ่มบุคคลที่เกี่ยวข้อง (Analysis of interest groups)

การวิเคราะห์กลุ่มบุคคลที่เกี่ยวข้อง เป็นการค้นหากลุ่มบุคคลที่เกี่ยวข้องหรือได้รับผลกระทบจากปัญหาในตารางปัญหา โดยจะเสนอในรูปแบบของรายการกลุ่มบุคคลที่เกี่ยวข้อง (List of interest group) ซึ่งประกอบด้วย

- หมายเลขประจำกลุ่ม ที่ขึ้นต้นด้วยอักษร 'I' และตามด้วยตัวเลข
- หมายเลขอ้างอิงปัญหา ที่มีผลกระทบต่อกลุ่ม
- หมายเลขอ้างอิงกิจกรรม ที่เกี่ยวข้องกับกลุ่มบุคคลและปัญหา

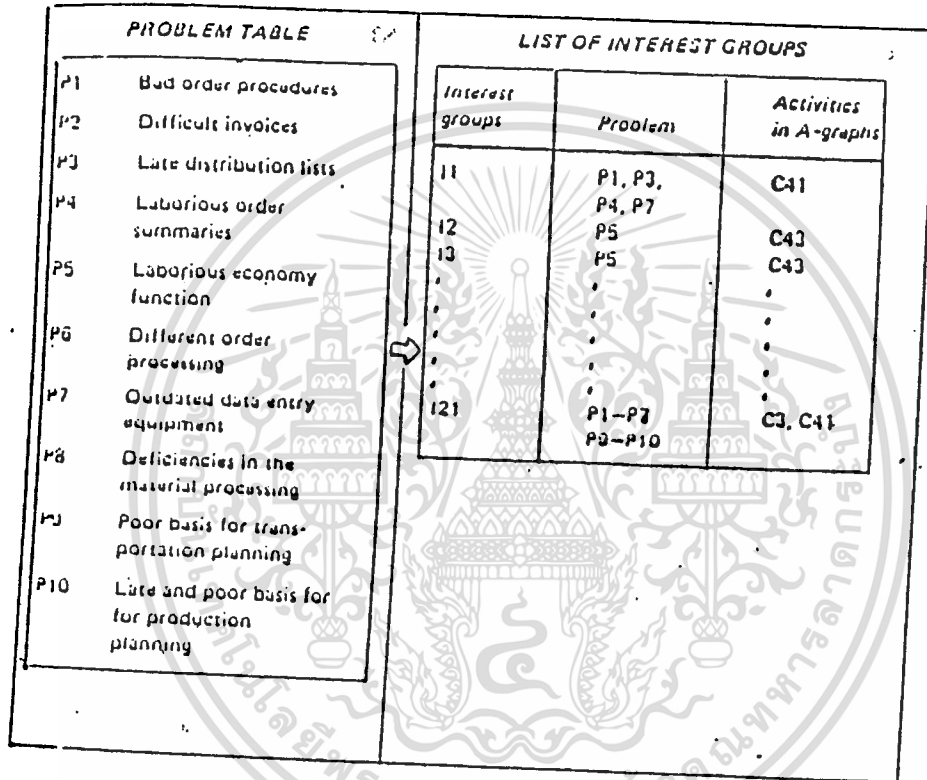
ในระหว่างที่ทำการวิเคราะห์กลุ่มที่สนใจ อาจเกิดกลุ่มที่สนใจกลุ่มใหม่ขึ้นมาพร้อมกับปัญหาใหม่ขึ้นได้ ซึ่งเมื่อเกิดกรณีเช่นนี้ขึ้นแล้วจะต้องทำการปรับปรุงในตารางปัญหาเดิมให้ถูกต้องด้วย

3. จัดกลุ่มปัญหา (Problem grouping)

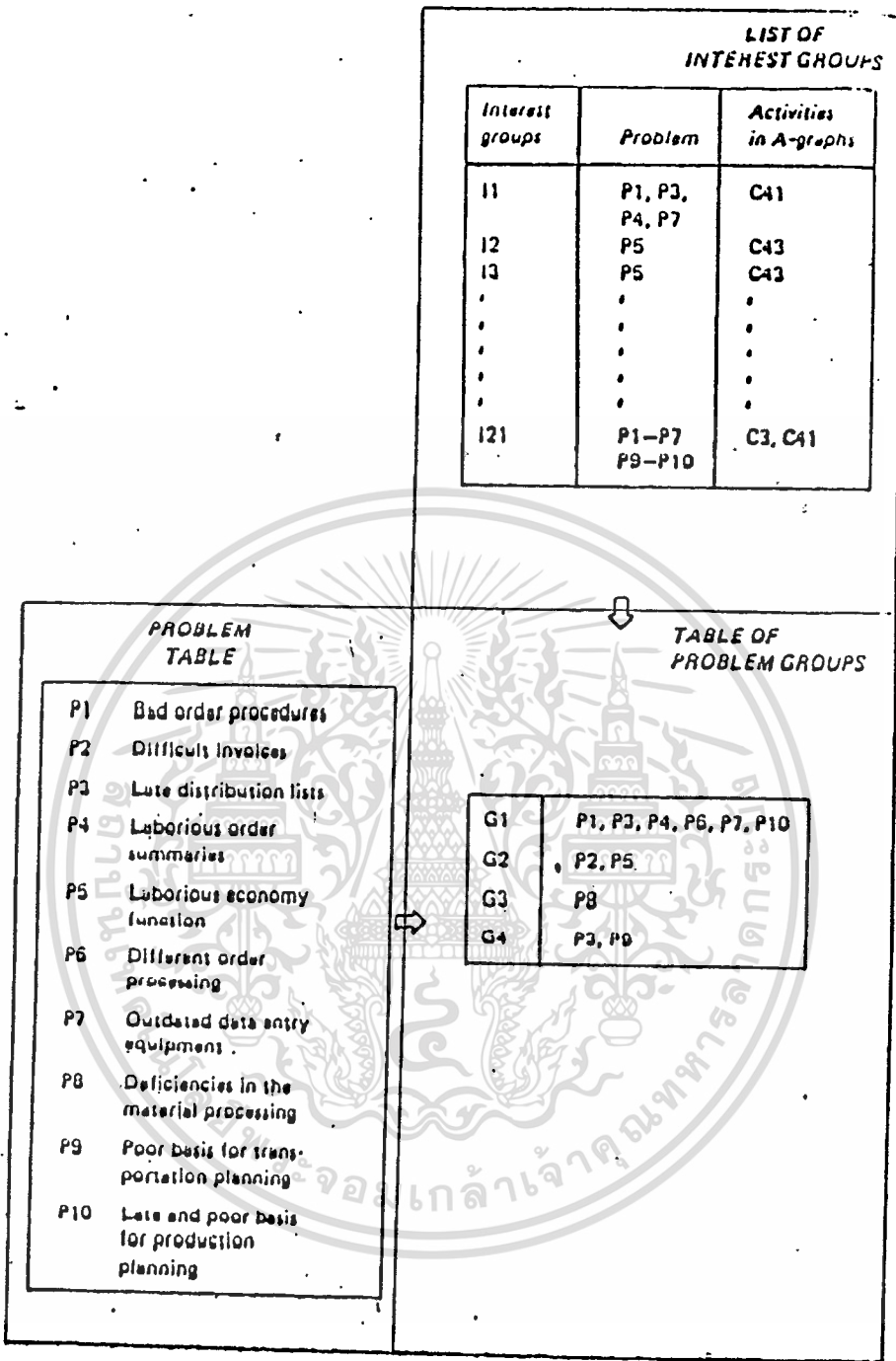
การที่จะทำงานกับปัญหาจำนวนมากในเวลาเดียวกันนั้น เป็นเรื่องที่ยุ่งยากลำบาก ดังนั้นปัญหาไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หาในตารางปัญหาซึ่งถูกแบ่งออกเป็นกลุ่ม ๆ และนำเลขเอในรูปของ ตารางกลุ่มปัญหา (Problem group table) ที่ประกอบด้วย

- หมายเลขอ้างอิงกลุ่มปัญหา ที่ขึ้นต้นด้วย 'G' และตามด้วยหมายเลขกำกับ
- หมายเลขอ้างอิงปัญหาที่อยู่ในกลุ่มนั้น

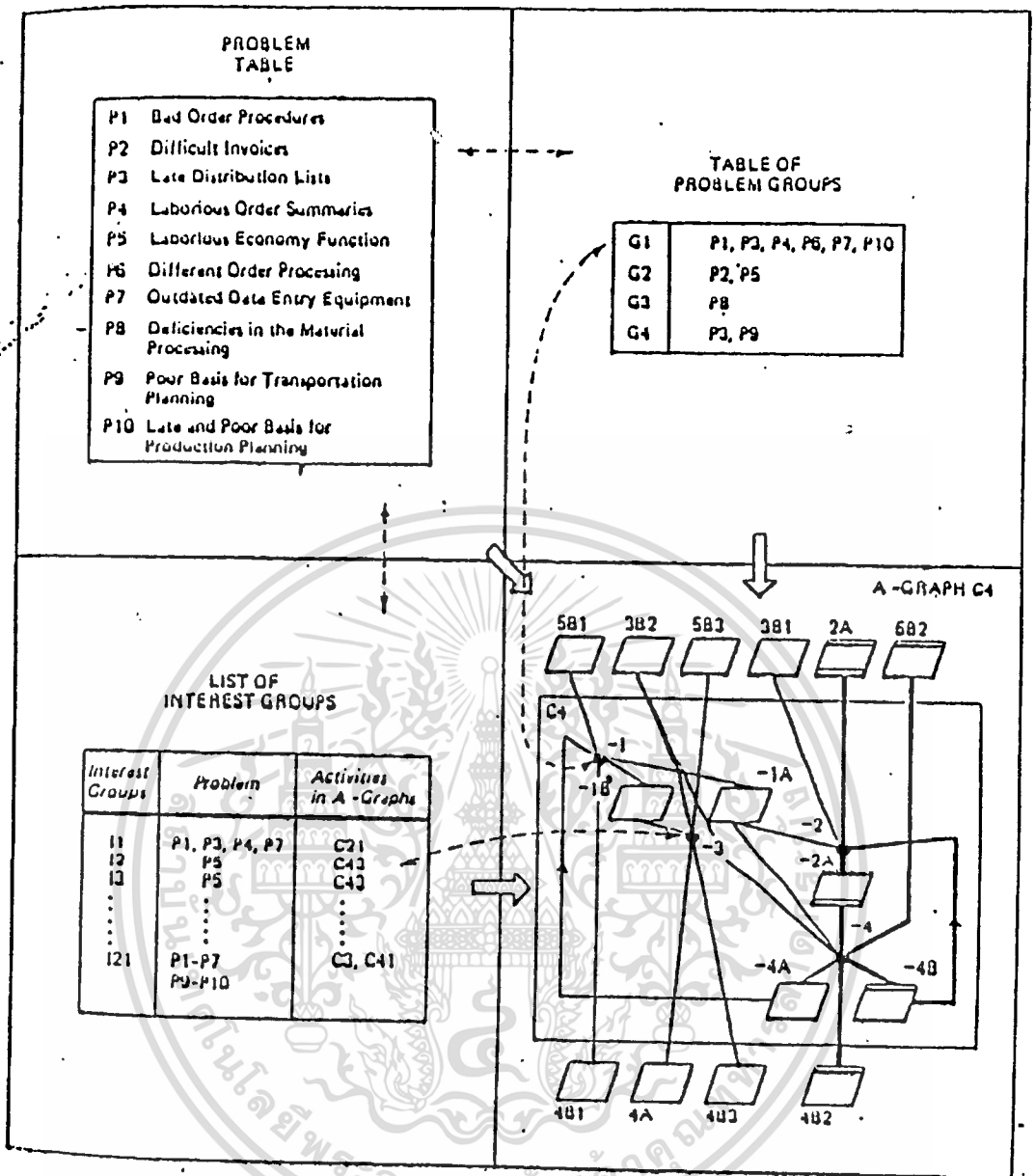


รูปที่ 2.8 การวิเคราะห์กลุ่มบุคคลที่เกี่ยวข้อง



รูปที่ 2.9 การจัดกลุ่มปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 การให้รายละเอียดกิจกรรมที่ทำในปัจจุบัน

4. ให้อายละเอียดกิจกรรมที่กำลังทำในปัจจุบัน (Description of current activities)

เป็นการแสดงกิจกรรมที่กำลังทำในปัจจุบันที่มีความความสัมพันธ์เกี่ยวข้องกับปัญหาและกลุ่มบุคคลที่เกี่ยวข้อง ซึ่งตรงนี้ควรที่จะให้ทุกกลุ่มบุคคลที่เกี่ยวข้องได้ให้รายละเอียดในส่วนที่ตนเองหรือเกี่ยวข้องกับอยู่ และเมื่อมีความเห็นแตกต่างกันระหว่างกลุ่มก็อาจจะต้องมีการทำความเข้าใจ

5. การกำหนดวัตถุประสงค์ของงานที่กำลังทำ (Description of objectives)

โดยปกติกลุ่มบุคคลที่เกี่ยวข้องแต่ละกลุ่มต่างก็มีความต้องการที่แตกต่างกันออกไป ซึ่งความต้องการนี้อาจจะขัดแย้งกันก็ได้ ดังนั้นจะต้องทำการสรุปวัตถุประสงค์เพื่อแก้ปัญหาต่าง ๆ ออกมาให้ตรงความต้องการ โดยจะต้องวัตถุประสงค์ที่เป็นไปได้ และแสดงรายละเอียดของแต่ละวัตถุประสงค์ให้ชัดเจน ผลที่ได้จากการทำงานในขั้นนี้จะแสดงในรูปของ ตารางวัตถุประสงค์ (table of objectives) ที่ประกอบด้วย

- หมายเลขเป้าหมาย ที่แทนด้วยตัวเลข
- เป้าหมาย

6. ประเมินความเหมาะสมในปัจจุบัน (Evaluation of current situation)

ในขั้นนี้เป็นการเปรียบเทียบสิ่งที่ต้องการ (วัตถุประสงค์) กับสิ่งที่มีอยู่ (ตารางปัญหาและกิจกรรมที่กำลังทำในปัจจุบัน) ผลที่ได้เป็นการเปลี่ยนจากปัญหาไปเป็น ความต้องการสำหรับการเปลี่ยนแปลง (need for changes) ความต้องการสำหรับการเปลี่ยนแปลงนี้จะถูกบันทึกลงใน ตารางความต้องการสำหรับการเปลี่ยนแปลง (table of need for changes) โดยแต่ละตารางจะสำหรับแต่ละกลุ่มปัญหา ซึ่งตารางนี้จะถูกใช้เป็นพื้นฐานในการศึกษาแนวทางต่าง ๆ ในการเปลี่ยนแปลงต่อไป

ศึกษาแนวทางต่าง ๆ ในการเปลี่ยนแปลง (Study of Change Alternatives)

การศึกษาแนวทางต่าง ๆ ที่จะใช้เปลี่ยนแปลง จะกระทำกับแต่ละกลุ่มปัญหา โดยใช้วิธีดังต่อไปนี้

1. สร้างแนวทางต่าง ๆ ที่จะใช้เปลี่ยนแปลง (Generation of change alternatives)
เป็นการคิดหาแนวทางการเปลี่ยนแปลงที่เป็นไปได้ทั้งหมด ที่สามารถช่วยแก้ปัญหาที่เกิดขึ้นกับระบบงานได้ งานส่วนนี้ไม่มีขั้นตอนอะไรแต่ขึ้นอยู่กับประสบการณ์ของแต่ละคน แนวทางที่จะใช้เปลี่ยนแปลงที่ได้มาจะถูกบันทึกไว้ใน ตารางแนวทางที่จะใช้เปลี่ยนแปลง (table of change alternatives)

2. ให้อายละเอียดแนวทางการเปลี่ยนแปลง (Description of change alternatives)

ในขั้นนี้เป็นการวิเคราะห์และประเมินผลที่ได้จากแต่ละแนวทางที่จะใช้เปลี่ยนแปลง โดยแสดงอยู่ในรูปของโมเดลกิจกรรม ที่ประกอบด้วย แผนภูมิกิจกรรม, รายการกำกับ และตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงคุณลุ่มนัต

3. ประเมินแต่ละแนวทางการเปลี่ยนแปลง (Evaluation of change alternatives)

เป็นการเปรียบเทียบข้อดีและข้อเสียของแนวทางที่จะใช้เปลี่ยนแปลงที่มีต่อ มนุษย์, สังคม และเศรษฐกิจว่าเป็นอย่างไรบ้าง คุ่มหรือไม่ว่าจะทำการเปลี่ยนแปลงนั้น โดยพิจารณาจากแต่ละกลุ่มบุคคลที่เกี่ยวข้อง ผลที่ได้จะใช้เป็นพื้นฐานในการเลือกแนวทางสำหรับเปลี่ยนแปลงต่อไป

เลือกแนวทางในการเปลี่ยนแปลง (Choice of Change Approach)

งานในส่วนสุดท้ายของการวิเคราะห์การเปลี่ยนแปลงก็คือ การเลือกแนวทางสำหรับเปลี่ยนแปลง โดยใช้วิธีดังต่อไปนี้

1. เลือกแนวทางในการเปลี่ยนแปลง

การตัดสินใจเลือกแนวทางในการเปลี่ยนแปลง (ต่อกลุ่มปัญหา) มีหลักเกณฑ์ดังนี้

- ทำการประเมินค่าทางสังคม และทางเศรษฐกิจ
- เข้าถึงปัจจัยที่มีความจำเป็นต่อการพัฒนา

ในขั้นตอนนี้เมื่อได้ทำการเลือกแล้ว สำหรับแนวทางใดที่ไม่ได้เลือกใช้ควรมีเอกสารเก็บรายละเอียดเอาไว้ และถ้ามีความเห็นขัดแย้งในกลุ่มบุคคลที่เกี่ยวข้องหลายๆ ก็ไม่ควรจะรีบเร่งเลือกแนวทางเปลี่ยนแปลง เพราะอาจจะทำให้งานล้มเหลวได้

2. เลือกวิธีการในการพัฒนา (Choice of development measures)

ในแนวทางการเปลี่ยนแปลงที่เลือกใช้ จะประกอบด้วยวิธีการในการพัฒนาหลายวิธี เช่น การพัฒนาระบบสารสนเทศ, การพัฒนาของกิจกรรมที่เกี่ยวข้องโดยตรง การพัฒนาองค์การ การพัฒนาบุคคล เป็นต้น

3. วิเคราะห์วิธีการพัฒนาที่ขนานกัน (Analysis of parallel development measure)

การวิเคราะห์ผู้กระทำ เพื่อดูว่าการปฏิบัติในการพัฒนาที่ขนานกันมีผลกระทบต่อกันอย่างไรตามระดับต่อไปนี้

- ความแตกต่างของการปฏิบัติในการพัฒนาภายในกลุ่มปัญหาเดียวกัน
- ความสัมพันธ์ระหว่างกลุ่มปัญหาภายในขอบเขตของกิจกรรมเดียวกัน
- สิ่งแวดล้อมที่มีผลกระทบต่อขอบเขตของกิจกรรมอื่น ๆ

หมายเหตุ

ขั้นตอนต่างในไอแอลมีรายละเอียดอีกมาก ควรที่ผู้สนใจจะศึกษาเพิ่มเติมจากเอกสารอ้างอิงท้ายเล่มวิทยานิพนธ์ฉบับนี้ นอกจากนี้ขอให้เข้าใจในที่นี้ด้วยว่าโครงการพัฒนาระบบสารสนเทศที่ได้ทำนี้ เพียงแต่นำบางส่วนของขั้นตอนของไอแอลมาใช้เท่านั้น และในส่วนที่ใช้ไม่ได้ใช้เต็มรูปแบบนัก) เครื่องมือสำคัญที่ได้นำมาใช้ก็คือ เอ-กราฟ และรายงานกำกับเป็นหลัก ดังนั้นจะไม่สามารถกล่าวอ้างได้ว่าการพัฒนาระบบที่ได้ทำการพัฒนาไปนี้ ได้พัฒนาไปในแนวทางของไอแอล หากแต่ให้นำเอาวิธีการของไอแอลมาประยุกต์ใช้เท่านั้น

2.2 ระบบฐานข้อมูล

2.2.1 ความหมายและที่มา

ฐานข้อมูล(Database) คือ แหล่งเก็บข้อมูล และความสัมพันธ์ระหว่างฐานข้อมูลเหล่านี้
[หนังสืออ้างอิง 9]

ในอดีตที่เน้นการจัดเก็บและค้นหาข้อมูลยังเก็บอยู่ในรูปแบบของไฟล์ (File) ซึ่งก็ใช้งาน
ได้ในระบบงานเล็ก ๆ มีผู้ใช้งานอยู่เพียงหนึ่งคนหรือไม่กี่คน แต่ในระบบที่มีขนาดใหญ่ขึ้นมีผู้ใช้งาน
การค้นหาและจัดเก็บข้อมูลจึงต้องการประสิทธิภาพที่สูงขึ้น สาเหตุที่ทำให้มีการนำระบบฐานข้อมูล
มาใช้แทนระบบไฟล์แบบเก่ามีดังนี้

- ลดความซ้ำซ้อนของข้อมูล(Minimum Redundancy)

การที่แผนกต่าง ๆ ต้องการข้อมูลชนิดเดียวกัน แต่ต่างกันแยกเก็บทำให้สิ้นเปลืองเนื้อที่
ในการจัดเก็บข้อมูลหน่วยความจำ

- หลีกเลี่ยงการขัดแย้งกันเองของข้อมูล (Data Inconsistency)

เมื่อมีการเปลี่ยนแปลงข้อมูลที่จัดเก็บในหลายที่หลายไฟล์ จะต้องตามทำการเปลี่ยนแปลง
ทุกไฟล์ที่จัดเก็บ ถ้าทำไม่ครบทุกไฟล์ก็จะทำให้ข้อมูลมีความผิดพลาดไปจากความจริงและขัดแย้งกัน

- แก้ปัญหาการรักษาความปลอดภัยของข้อมูล(Data Security Problem)

เนื่องจากการเก็บข้อมูลกระจัดกระจาย ทำให้ยากต่อการรักษาความปลอดภัยเพื่อการ
ซ้ำไหลและดัดแปลงข้อมูล

- ทำให้ข้อมูลมีบูรณภาพมากที่สุด(Maximum Integrity) หรือการควบคุมความถูกต้อง
ของข้อมูล

- ทำให้สามารถควบคุมการใช้งานโดยส่วนกลางได้(Centralize Control)

- สามารถใช้ข้อมูลร่วมกันได้

- มีความคล่องตัวและความยืดหยุ่นในการใช้งาน(Data Independent)

- มีการควบคุมมาตรฐานร่วมกัน

2.2.2 ระบบจัดการฐานข้อมูล(Database Management System หรือ DBMS)

คือซอฟต์แวร์ที่เปรียบเสมือนสื่อกลางระหว่างผู้ใช้และโปรแกรมต่างๆที่เกี่ยวข้องกับการใช้
ฐานข้อมูล [หนังสืออ้างอิง 4] หรือคือซอฟต์แวร์ที่กำหนดวิธีการการเข้าถึง(access)ทุกอย่าง
สู่ฐานข้อมูล [หนังสืออ้างอิง 11]

เนื่องจากการใช้และควบคุมดูแลฐานข้อมูลเป็นเรื่องที่ซับซ้อนยุ่งยาก ระบบจัดการฐานข้อมูล
จึงเป็นเครื่องมือสำคัญในการที่ลดภาระของผู้ใช้ไปอย่างมาก ทำให้การใช้งานระบบฐานข้อมูลมี
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่โดยไม่ขออนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประสิทธิภาพสูงขึ้น

2.2.3 โมเดลของข้อมูล (Data Model)

ให้แก่ประเภทของระบบการจัดการฐานข้อมูล แบ่งออกโดยจำแนกตามการแสดงความสัมพันธ์ระหว่างข้อมูลได้เป็นสามโมเดลดังนี้

1. แบบโครงข่าย (Network Model)

เป็นการแสดงความสัมพันธ์ด้วยลิงก์ลิสต์เป็นโครงข่าย โดยมีตัวชี้ (pointer) ซึ่งระหว่างข้อมูลเชื่อมกันเป็นชุด มีความสัมพันธ์แบบ แมทีทูแมนี (many to many)

2. แบบแผนภูมิต้นไม้ (Hierarchical Model)

ทำการแทนความสัมพันธ์ระหว่างข้อมูลในลักษณะของแผนภูมิต้นไม้ (tree) โดยมีความสัมพันธ์ในลักษณะ วันทูแมนี (one to many)

3. แบบสัมพันธ์ (Relational Model)

เป็นการเก็บข้อมูลที่มีลักษณะการเก็บในรูปแบบตาราง (table) 2 มิติธรรมดา คือมีแถว (row), คอลัมน์ (column)

2.2.4 สถาปัตยกรรมของระบบฐานข้อมูล (Architecture for a Database System)

ลักษณะสถาปัตยกรรมของระบบฐานข้อมูล (Data Base System) ได้ถูกกำหนดให้เป็นมาตรฐานจาก 3 องค์การหลัก คือ ISO (International Standard Organization), IFIP (International Federation for Information Processing) และ ANSI (American National Standard Institute) โดยสามารถจัดแบ่งออกเป็น 3 ระดับด้วยกันคือ

1. นิยามข้อมูลระดับภายนอก (External Schema)

2. นิยามข้อมูลระดับแนวคิด (Conceptual Schema)

3. นิยามข้อมูลระดับภายใน (Internal Schema)

ทั้งสามระดับมีความสัมพันธ์กันดังในภาพที่ 2.11 โดยรายละเอียดของส่วนต่าง ๆ มีดังนี้

1. ผู้ใช้ (User)

• ผู้ใช้งานที่สิ้นสุดท้าย (End-user), คนเขียนโปรแกรม, โปรแกรมใช้งาน

2. นิยามข้อมูลระดับภายนอก

• โครงสร้างข้อมูลที่ผู้ใช้แต่ละคนมองเห็น

3. นิยามข้อมูลระดับแนวคิด

• เป็นส่วนกำหนดลักษณะ, ขนาดและโครงสร้างของข้อมูล และความสัมพันธ์ระหว่างข้อมูลทั้งหมดที่อยู่ในขอบเขตของงาน

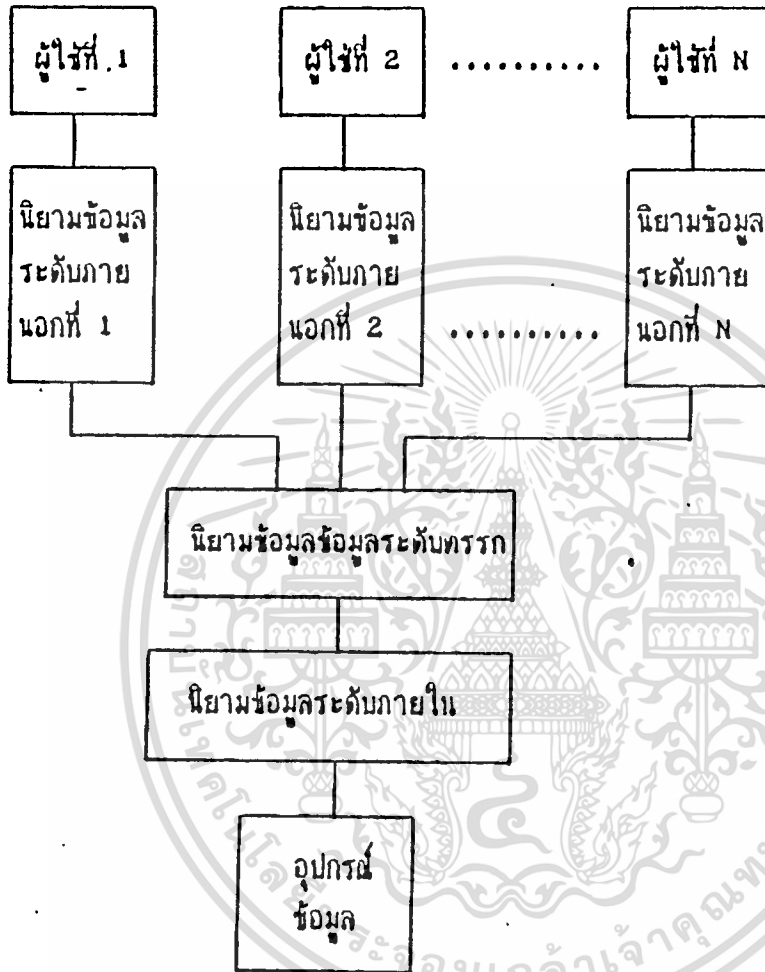
4. นิยามข้อมูลระดับภายใน

• โครงสร้างข้อมูลที่จัดเก็บในอุปกรณ์เก็บข้อมูล ลักษณะการเก็บข้อมูล

5. ฐานข้อมูลทางกายภาพ

• อุปกรณ์เก็บข้อมูล (เช่น ฮาร์ดดิสก์)

(Physical database)

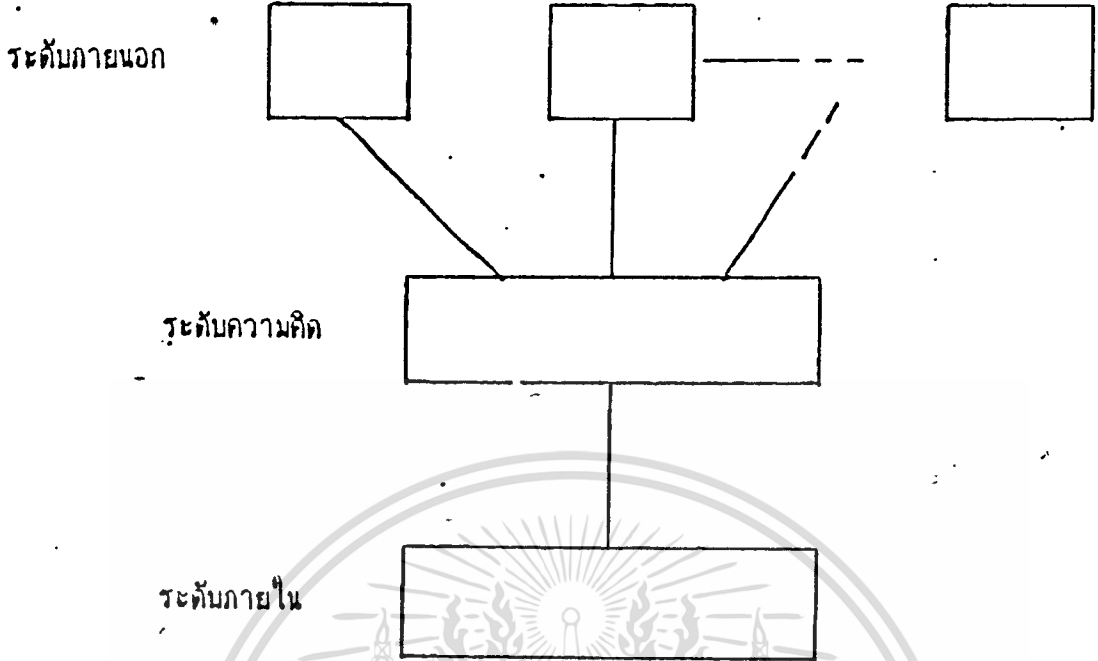


รูปที่ 2.11 สถาปัตยกรรมของระบบฐานข้อมูล

หรืออาจมองสถาปัตยกรรมดังกล่าวออกเป็นระดับๆ ได้ดังนี้

- 1 ระดับกายภาพ หรือ ระดับภายใน (Physical level, Internal level)
- 2 ระดับแนวคิด หรือ ระดับตรรก (Conceptual level, Logical level)
- 3 ระดับภายนอก หรือ ระดับผู้ใช้ (External level, User view)

ซึ่งแสดงไว้ในภาพที่ 2.12



รูปที่ 2.12 ระดับต่างๆของสถาปัตยกรรมของระบบฐานข้อมูล

2.2.5 ผู้ใช้

ในระบบฐานข้อมูลได้แบ่งผู้ใช้ออกเป็นสามกลุ่มดังนี้

1. ผู้ใช้งานขั้นสุดท้าย (End-user)

ได้แก่ผู้ที่จะได้รับข่าวสารที่เหมาะสมตามชนิดของงาน และความต้องการของตนจากฐานข้อมูล โดยทั่วไปจะเป็นผู้ที่มีความรู้ทางคอมพิวเตอร์และฐานข้อมูลน้อยมาก

2. คนเขียนโปรแกรมใช้งานฐานข้อมูล (Application Programmer)

จะเป็นผู้ที่เขียนโปรแกรมใช้งานฐานข้อมูลให้เป็นไปตามความต้องการของผู้ใช้งานขั้นสุดท้าย

3. ผู้บริหารฐานข้อมูล (DBA : Data Base Administrator)

เป็นผู้ที่ทำหน้าที่รับผิดชอบควบคุมระบบฐานข้อมูลทั้งหมด โดยมีคุณสมบัติหรือหน้าที่ดังนี้

1 เป็นผู้เชี่ยวชาญทางด้านเทคนิคระดับสูงและใช้ระบบจัดการฐานข้อมูลเป็น

2 เป็นผู้ออกแบบนิยามข้อมูลระดับแนวความคิดของทั้งระบบงาน

3 เป็นผู้จัดการนิยามข้อมูลระดับภายนอกให้แก่ผู้ใช้ขั้นสุดท้ายแต่ละคน

รวมทั้งการให้อำนาจ

(Authority) ที่เหมาะสมแก่ผู้ใช้ขั้นสุดท้าย

4 เป็นผู้พิจารณาเลือกทฤษฎีการเข้าถึงข้อมูล (Access Method) ที่เหมาะสมรวมทั้งอุปกรณ์

ที่จะใช้ในการจัดเก็บข้อมูลด้วย

5 เป็นผู้จัดการปรับปรุงการทำงาน (Tune Performance) ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 6 เป็นผู้กำหนดรูปแบบในการตรวจสอบความถูกต้องแน่นอนของข้อมูล
- 7 เป็นผู้กำหนดวิธีการในการเก็บข้อมูลสำรอง (back up) และการนำกลับมาใช้ใหม่ (recovery)
- 8 เป็นผู้คอยติดต่อผู้ใช้ในสุดท้าย เพื่อให้การทำงานของผู้ใช้ในสุดท้ายทำไปได้โดยมีประสิทธิภาพตรงตามความต้องการ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ระบบจัดการฐานข้อมูลแบบสัมพันธ์ ออเรเคิล

2.3.1 ออเรเคิล

ออเรเคิลเป็นระบบจัดการฐานข้อมูลแบบสัมพันธ์ตัวหนึ่งซึ่งมีประสิทธิภาพสูง ผลิตและพัฒนาขึ้นโดยบริษัท Oracle Corporation มีเวอร์ชันให้ใช้ได้ตั้งแต่เครื่องไมโครคอมพิวเตอร์ ไปจนถึงเครื่องระดับเหนือไมโครขึ้นไป โปรแกรมมีความสามารถในการสร้างฐานข้อมูลสำหรับเก็บข้อมูลที่ต้องการ โดยอยู่ในรูปของตารางตามคุณสมบัติของฐานข้อมูลแบบสัมพันธ์และสามารถแก้ไขเปลี่ยนแปลง หรือค้นหาข้อมูลตามเงื่อนไขที่กำหนดโดยผ่านภาษาซีเคิล การทำงานกับฐานข้อมูลอาจทำโดยใช้ภาษาซีเคิลโดยตรง หรือใช้ภาษาคอมพิวเตอร์ภาษาอื่นๆ เรียกใช้ซีเคิล เฉพาะในส่วนที่เกี่ยวกับฐานข้อมูลก็ได้ นอกจากนี้ออเรเคิลยังมีส่วนอำนวยความสะดวกสำหรับงานต่างๆ อีกมากมาย เช่น การสร้างเมนู การสร้างแบบฟอร์มสำหรับป้อนข้อมูลเข้า การจัดทำรายงาน เป็นต้น ในปัจจุบันออเรเคิลมียอดขายสูงมาก ผลิตกันที่สร้างผลกำไรแก่บริษัทผู้ผลิตอย่างมากมาย ในด้านตลาดจะเป็นรองอยู่ที่แต่ ดีเบส (DBASE) ของ Ashton-Tate เท่านั้น สิ่งเหล่านี้เป็นเครื่องยืนยันเป็นอย่างดีถึงการเป็นที่ยอมรับของออเรเคิลในแวดวงการใช้ระบบจัดการฐานข้อมูล ในแง่ของประสิทธิภาพในความเป็นระบบจัดการฐานข้อมูลแบบสัมพันธ์

2.3.2 การนิยามรากศัพท์ของออเรเคิลในฐานะระบบจัดการฐานข้อมูลแบบสัมพันธ์

EDGAR F. CODD ได้เสนอเกณฑ์ 12 ข้อในการเปรียบเทียบวิเคราะห์คุณสมบัติของซอฟต์แวร์ที่เป็นระบบจัดการฐานข้อมูลแบบสัมพันธ์ ในฐานะที่ EDGAR F. CODD ได้เป็นผู้ริเริ่มเสนอฐานข้อมูลแบบสัมพันธ์ตั้งแต่ปี 1970 ซึ่งถือว่าเป็นผู้บุกเบิกวิชาการทางด้านนี้โดยเฉพาะ หลักเกณฑ์ทั้ง 12 ข้อนี้จึงถูกใช้เป็นมาตรฐานในการวัดความเป็นระบบจัดการฐานข้อมูลแบบสัมพันธ์ของบรรดาซอฟต์แวร์ระบบจัดการฐานข้อมูลแบบสัมพันธ์ต่างๆ หนึ่งคืออ้างอิง 1-3 การศึกษากฎเกณฑ์ดังกล่าวจึงถือว่าเป็นการศึกษาระบบจัดการฐานข้อมูลแบบสัมพันธ์ไปในตัวด้วย

รายละเอียดของหลักเกณฑ์ 12 ข้อและการพิจารณา ออเรเคิลภายใต้หลักเกณฑ์เหล่านี้มีดังนี้คือ

กฎข้อที่ 1 กฎข่าวสาร (The Information Rule)

เป็นกฎว่าด้วยรูปแบบของข้อมูล กล่าวไว้ว่าข้อมูลที่เก็บในฐานข้อมูลแบบสัมพันธ์ ในระดับตรรก จะต้องถูกเก็บในรูปตารางเท่านั้น ซึ่งในออเรเคิลเองก็มีคุณสมบัติในข้อนี้

กฎข้อที่ 2 กฎการรับประกันการเข้าถึงข้อมูล (Guarantee Access Rules)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้โดยไม่ขึ้นด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กล่าวได้ว่าผู้ใช้ สามารถเข้าถึงข้อมูลทุกหัวในตารางได้ด้วยการระบุชื่อตาราง ค่าของไพล์
เมะริคีย์ (primary key และ คอลัมน์ (column) ที่ต้องการ และหากไม่มีการเปลี่ยนแปลงข้อมูล
ในตารางแล้ว ผลของคำสั่งเดียวกัน ย่อมจะได้ออกมาเหมือนกันทุกครั้งที่ไป สำหรับออเรเคิล เอง
ก็มีลักษณะเช่นนี้ ในออเรเคิลสามารถกำหนด ยูนิค อินเดค (unique index) กับ ไพล์เมะริคีย์
ทำให้ไม่มีค่าไพล์เมะริคีย์ที่ซ้ำกัน ทำให้การเข้าถึงข้อมูลทุกหัวในตารางโดยไพล์เมะริคีย์ สามารถ
กระทำได้

กฎข้อที่ 3 วิธีการแสดงว่าข้อมูลบางรายการยังไม่พร้อมหรือยังไม่มีค่า (Systematic Treatment of Missing Information)

ว่าด้วยวิธีแสดงข้อมูลในส่วนที่ไม่พร้อมหรือไม่มีค่า (หรือไม่ทราบค่า) โดยแสดงด้วยค่า null ซึ่งความหมายหรือค่าของ null นี้จะแตกต่างไปจากค่า 0 หรือ ช่องว่าง (blank) ทั้งนี้
ออเรเคิลยังมีฟังก์ชันที่สามารถเปลี่ยนค่าของ null เป็นค่าอื่นได้ (เป็นค่าชั่วคราวเท่านั้น) เช่น ให้มีค่า
เท่ากับ 0 เป็นต้น

กฎข้อที่ 4 โครงสร้างแคตตาล็อกจะต้องมีสภาพโมเดลแบบสัมพันธ์ (Dynamic on-line catalog must be base on relational model)

ก่อนอื่นขอเริ่มด้วยความหมายของแคตตาล็อก (catalog) ก่อน แคตตาล็อกก็คือส่วนหนึ่ง
ของระบบจัดการฐานข้อมูลที่ไว้เก็บโครงสร้างของข้อมูลอันได้แก่ตาราง ชื่อคอลัมน์ รายชื่อผู้มีสิทธิ
ใช้ตาราง เป็นต้น รายละเอียดของกฎมีอยู่ว่า รูปแบบของข้อมูลในแคตตาล็อก เองก็ต้องอยู่ในรูป
แบบของฐานข้อมูลแบบสัมพันธ์ และการปฏิบัติใดๆที่จะมีกับข้อมูลในแคตตาล็อกก็จะเป็นไปในลักษณะ
เดียวกับหรือวิธีการเดียวกับที่กำกับข้อมูลของระบบ ในออเรเคิลส่วนที่เป็นแคตตาล็อกได้แก่ พจนานุกรม
ของข้อมูล (data dictionary) ซึ่งประกอบด้วยกลุ่มของตารางและวิว โดยครอบคลุม
โครงสร้างฐานข้อมูลทั้งหมด ทุกครั้งที่มีการเปลี่ยนแปลงในฐานข้อมูล ก็จะมีการเปลี่ยนแปลงข้อมูล
ในพจนานุกรมของข้อมูลตามไปด้วย

กฎข้อที่ 5 กฎการใช้ภาษา (Comprehensive data sublanguage rule)

กล่าววาระบบจัดการฐานข้อมูลแบบสัมพันธ์จะต้องมีภาษาคำควบคุมข้อมูล (Data Manipulation Language) ที่อย่างน้อยมีความสามารถในการกระทำฟังก์ชันพีชคณิตแบบสัมพันธ์ (Relational Algebra) ต่อไปนี้คือ ซีเล็ค (select) โปรเจค (project) จอยน์ (join)
สำหรับออเรเคิลนั้นใช้ภาษา ซีควอลเป็นภาษาคำควบคุมข้อมูล ซึ่งสามารถทำสิ่งเหล่านี้ได้ครบ
ดังนั้นออเรเคิลจึงเป็นไปตามหลักเกณฑ์ข้อนี้

กฎข้อที่ 6 กฎการแก้ไขข้อมูลผ่านวิว (View updating rule) อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พูดกันว่าเมื่อมีการสร้างวิว การควบคุมการแก้ไขข้อมูลผ่านวิว ย่อมจะต้องมีการพิจารณาว่า
สิ่้จุม่ารกดกระทำไ้แก่ไ้ไหน จะมีผลทำให้สูญเสียความถูกต้องในตารางจริงหรือไม่ เป็นหน้าที่ของ
ระบบจัดการฐานข้อมูลที่จะต้องเข้ามาควบคุม โดยทั่วไปแล้วการแก้ไขข้อมูลในวิวที่สร้างจากตา
รางเดี่ยวในระบบจัดการฐานข้อมูลแบบสัมพันธ์หลายตัวมักไม่มีปัญหา(รวมทั้ง ออเรเคิล) แต่สำ
หรับวิวที่สร้างจากตารางมากกว่า 1 ตารางแล้ว การแก้ไขข้อมูลในวิว จะมีปัญหายุ่มาก ตรงนี้
ออเรเคิล กำหนดเป็นข้อจำกัดว่าห้ามทำการแก้ไขข้อมูลกับวิว ที่ถูกสร้างในลักษณะนี้เลย (ไม่ยอม
ให้ทำ) จริงแล้วตามเกณฑ์ข้อนี้ สำหรับวิวที่ถูกสร้างขึ้นมาจากตารางใดหนึ่ง ถ้ามีการแก้ไขที่วิว
ระบบจัดการฐานข้อมูลจะต้องสามารถจัดการแก้ไขลงบนตารางจริงนั้นได้ด้วย ซึ่งตรงนี้จะเห็นว่า
ออเรเคิลกระทำไ้ได้โดยมีข้อจำกัดดังกล่า

กฎข้อที่ 7 ความสามารถในการเพิ่ม ลด และแก้ไขข้อมูล (High-Level insert
update, and delete)

ว่าด้วยความสามารถในการแก้ไข เพิ่ม ลดข้อมูลโดยกระทำในระดับของเซต(set-level
Operation) เนื่องจากใช้ออเรเคิลใช้ซีเคิล ซึ่งซีเคิลสามารถทำไ้ได้อยู่แล้วออเรเคิลจึงมีคุณ
สมบัติตามกฎข้อนี้

กฎข้อที่ 8 ความเป็นอิสระของข้อมูลในระดับกายภาพ (Physical data
independence)

พูดถึงความเป็นอิสระของข้อมูลทางกายภาพ คือในแง่ผู้ใช้แล้วไม่จำเป็นต้องรับรู้ถึงการจัด
เก็บข้อมูลจริง รวมทั้งการเปลี่ยนแปลงวิ้ก็ไม่มีผลกระทบกระเทือนกับการใช้งานที่ทำได้ สำหรับ
ออเรเคิลแล้วความเป็นอิสระของข้อมูลทางกายภาพอยู่ในระดับสูง โดยเฉพาะอย่างยิ่งกับผู้ใช้ในระ
ดับภายนอกนั้น ผู้ใช้ไม่จำเป็นต้องรับรู้ว่ามีโครงสร้างทางกายภาพ(Physical Structure)เช่น
ไรอยู่ในระบบ

กฎข้อที่ 9 ความเป็นอิสระของข้อมูลในระดับตรรก (Logical data independence)

พูดถึงความเป็นอิสระของข้อมูลในระดับตรรก การเปลี่ยนแปลงข้อมูลในระดับตรรกหมายถึง
การเปลี่ยนแปลงโครงสร้างของข้อมูล เช่น เพิ่มคอลัมน์ สร้างตารางใหม่ ขยายขนาดของคอลัมน์
เป็นต้น หากมีความเป็นอิสระของข้อมูลในทางตรรกแล้ว การเปลี่ยนแปลงโครงสร้างของข้อมูลไป
จากเดิมจะไม่มีผลต่องานที่ไ้กระทำไปแล้ว สำหรับออเรเคิลการเปลี่ยนแปลงขนาดของคอลัมน์ใน
ตารางที่มีข้อมูลอยู่แล้ว จะไม่สามารถกระทำไ้ได้ในทันทีจะต้องทำการลบข้อมูลเหล่านั้นออกจากตา
รางเสียก่อน ซึ่งจะเห็นว่าออเรเคิลมีความเป็นอิสระของข้อมูลในระดับตรรกเช่นกันแม้จะยุ่งยาก
สักเล็กน้อย

กฎข้อที่ 10 ความเป็นอิสระของบูรณภาพ (Integrity Independence)

มีหลักการที่ว่าระบบจัดการฐานข้อมูลแบบสัมพันธ์ควรจะสามารถจัดเก็บข้อมูลเกี่ยวกับการควบคุมบูรณภาพไว้ในแค็ตตาล็อกของระบบเป็นอิสระจากโปรแกรม โดยระบบจัดการฐานข้อมูลแบบสัมพันธ์เป็นตัวจัดการการควบคุมบูรณภาพ ผู้ใช้ไม่ต้องเขียนโปรแกรมเพื่อจัดการเรื่องนี้อย่างไรก็ตาม สำหรับออเรเคิลจะมีอยู่เฉพาะบูรณภาพเชิงที่ตี (entity integrity) เท่านั้น ไม่มีบูรณภาพแบบอ้างอิง (referential integrity)

กฎข้อที่ 11 ความเป็นอิสระของการกระจาย (Distribution Independence)

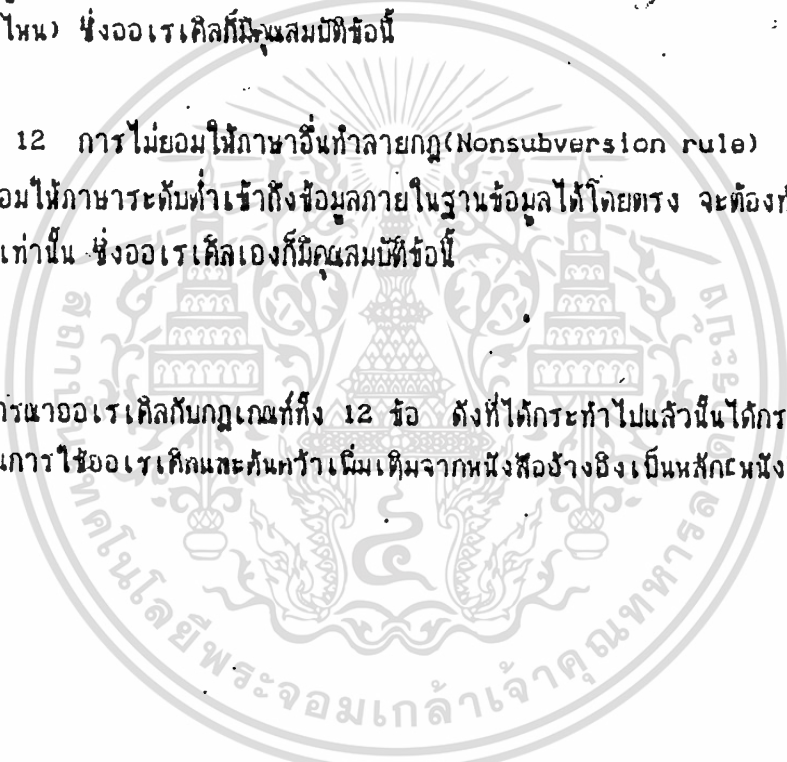
ความหมายมีอยู่ว่าถ้าระบบเดิมมีการย้ายคอมพิวเตอร์ไปติดตั้งที่อื่น หรือมีการนำระบบที่เดิมแยกกันอยู่เข้ามารวมเข้าด้วยกัน การใช้งานยังคงเป็นเช่นเดิม (ผู้ใช้ระบบไม่ต้องสนใจว่าข้อมูลถูกเก็บไว้ที่ไหน) ซึ่งออเรเคิลก็มีคุณสมบัติข้อนี้

กฎข้อที่ 12 การไม่ยอมให้ภาษาอื่นทำลายกฎ (Nonsubversion rule)

คือไม่ยอมให้ภาษาระดับต่ำเข้าถึงข้อมูลภายในฐานข้อมูลได้โดยตรง จะต้องทำผ่านระบบจัดการฐานข้อมูลเท่านั้น ซึ่งออเรเคิลเองก็มีคุณสมบัติข้อนี้

หมายเหตุ

การพิจารณาออเรเคิลกับกฎเกณฑ์ทั้ง 12 ข้อ ดังที่ได้กระทำไปแล้วนั้นได้กระทำโดยอาศัยประสบการณ์ในการใช้ออเรเคิลและค้นคว้าเพิ่มเติมจากหนังสืออ้างอิง เป็นหลักหนังสืออ้างอิง 13 ง



2.4 การออกแบบฐานข้อมูลแบบสัมพันธ์ในระดัประนาเกิดโดยอาศัยโมเดลโมเดล

2.4.1 การออกแบบฐานข้อมูล

โดยทั่วไปแล้วการออกแบบระบบฐานข้อมูล(database design) สามารถแบ่งเป็น 2 ระดับคือ

1. การออกแบบระดับข่าวสาร(INFORMATION-LEVEL DESIGN)
2. การออกแบบระดับกายภาพ(PHYSICAL-LEVEL DESIGN)

การออกแบบระดับข่าวสารนี้ จะประกอบด้วยการรวบรวมความต้องการของผู้ใช้, ข้อมูลต่างๆที่เกี่ยวข้อง, ความสัมพันธ์ของข้อมูล เพื่อสร้างโมเดลของข้อมูลและโครงสร้างของข้อมูลโดยที่นี้เรายังไม่สนใจถึง ประสิทธิภาพของระบบ(ซึ่งขึ้นกับระบบจัดการฐานข้อมูลที่ใช้)

การออกแบบระดับกายภาพ ประกอบด้วยการนำข้อกำหนดทางกายภาพของข้อมูล, ประสิทธิภาพในการทำงานอันได้แก่ ความถี่ในการแสดงรายงาน, ความเร็ว, การประมวลผล, ระบบการป้องกันภัยข้อมูล ฯลฯ เพื่อนำมาพิจารณาเลือกใช้ระบบจัดการฐานข้อมูลที่เหมาะสม เพื่อให้ได้ระบบฐานข้อมูลที่สมบูรณ์ในท้ายสุด

ในส่วนการออกแบบระดับข้อมูลได้มีผู้เสนอเกณฑ์ในการออกแบบฐานข้อมูลแบบสัมพันธ์ไว้ 3 ประการ ดังต่อไปนี้ข้ออ้างอิง 15 3

1. การเป็นตัวแทน (Representation) โครงสร้างที่ไว้ท้ายสุดจะต้อง จะต้องแสดงถึงข้อมูล ความสัมพันธ์ของข้อมูลและสิ่งเกี่ยวข้องเช่นเดิมไม่เปลี่ยนแปลง คือจะต้องไม่บิดเบือนข้อมูลเดิมเป็นอัน และ ไม่มีการขาดหายของข้อมูลส่วนหนึ่งส่วนใดไป (loss of information)

2. การแบ่งแยก (Separation) จากข้อมูลและความสัมพันธ์หรือข้อกำหนดพื้นฐาน จะต้องแบ่งแยกออกมาเป็นความสัมพันธ์ย่อยหลายความสัมพันธ์ โดยที่หากมองรวมแล้วยังต้องมีเกณฑ์ของการเป็นตัวแทนอยู่ การที่เราต้องแบ่งความสัมพันธ์ออกไปเช่นนี้มีสาเหตุอยู่คือ เพื่อให้โครงสร้างของข้อมูลไม่อยู่ในรูปแบบที่คอมพาว์ (COMPOUND) แต่ต้องการให้อยู่ในรูปแบบที่ง่าย (SIMPLE) ลักษณะฐานข้อมูลที่คอมพาว์ในโมเดลแบบสัมพันธ์อย่างที่เราเห็นง่ายคือการนำเอาทุกสิ่งทุกอย่างมาไว้ในตารางเดียวกัน ซึ่งก็ให้ผลเสียไม่แตกต่างไปจากการเก็บข้อมูลแยกไว้ต่างไฟล์ดังที่กล่าวหัวข้อ 2.2.1 (นั่นคือข้อมูลจะต้องเก็บแยกแต่จะต้องเก็บแยกโดยพิจารณาให้เหมาะสม)

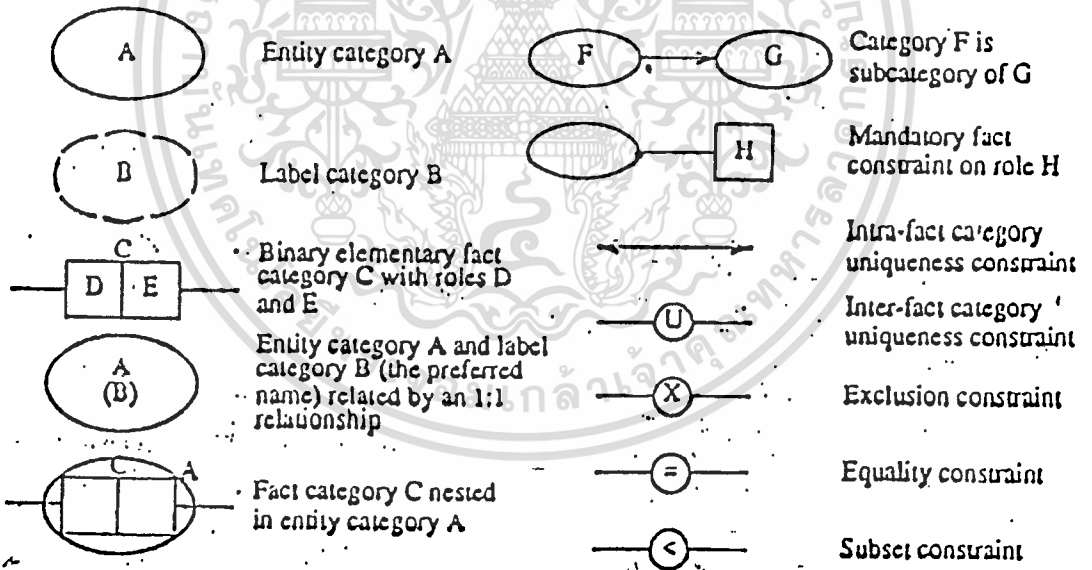
3. การซ้ำซ้อน (Redundancy) โครงสร้างฐานข้อมูลท้ายสุด จะต้องไม่มีข้อมูลซึ่งซ้ำซ้อน คำว่าซ้ำซ้อนคือการที่ข้อมูลชนิดเดียวกันถูกเก็บไว้หลายแห่ง (สำหรับโมเดลแบบสัมพันธ์ก็คือถูกเก็บไว้ในหลายตาราง) ในการประมวลผลซึ่งไม่ใช่ระบบฐานข้อมูลหากแต่ใช้ไฟล์ธรรมดา มักจะเกิดการซ้ำซ้อนเสมอทั้งนี้เพราะผู้ใช้แต่ละกลุ่มจะมีไฟล์ส่วนตัวเป็นของตนเองโดยบางส่วนของข้อมูลในไฟล์เหล่านั้นเป็นข้อมูลลักษณะเดียวกัน ซึ่งทำให้เกิดการซ้ำซ้อนขึ้นได้ จากความซ้ำซ้อนดังกล่าวมีผลต่อการแก้ไขข้อมูลซึ่งจะต้องทวนแก้ไขในส่วนที่ซ้ำให้ครบ ซึ่งเสี่ยงต่อความผิดพลาดได้ง่าย จาก

หลักเกณฑ์ข้อนี้มิได้อยู่ว่าจะไม่ยอมให้เกิดการซ้ำซ้อนขึ้นเลย ตรงนี้มิได้ให้ความเป็นชัดเจนไว้ว่าไม่สามารถจัดการซ้ำซ้อนได้หมด จะทำได้ก็เพียงลดการเกิดการซ้ำซ้อนได้เท่านั้นหนังสืออ้างอิง 4

11 3

2.4.2 โฉมของวิธีการวิเคราะห์ข้อมูลของโปเนน

ในโฉม ประกอบด้วย สิ่งที่เราสนใจ (Entity Type) เช่น ชื่อคน เพศ งานอดิเรก, สิ่งที่ใช้อ้างถึงสิ่งที่เราสนใจ (Label Type) เช่น นามสกุล, ความสัมพันธ์ระหว่างสิ่งต่างๆ (Facts) ซึ่งจะอยู่ในรูปของ ประธาน กริยา กรรม นอกจากนี้ยังมีข้อกำหนดต่างๆ คือ ข้อกำหนดของการเท่ากัน (Equality Constraint), ข้อกำหนดของความเป็นหนึ่ง (Uniqueness Constraint), ข้อกำหนดของความเป็นสับเซต (Subset Constraint), ข้อกำหนดช่วงความถี่ (Frequency Range Constraint), ข้อกำหนดในการที่เมื่อปรากฏข้อมูลอย่างหนึ่งต้องมีข้อมูลอีกอย่างหนึ่งด้วย (Total Role Constraint or Mandatory Role Position Constraint), สัญลักษณ์ของสิ่งต่างๆนี้แสดงในรูปที่ 2.13



รูปที่ 2.13 สัญลักษณ์ที่ใช้ในโฉม

กันอันเกิดจากความสัมพันธ์กันระหว่างแพลตฟอร์ม

ถ้าสิ่งที่เราสนใจถูกอ้างถึงได้มากกว่า 1 แบบจากข้างบน เราจะกล่าวว่าสิ่งที่เราสนใจนั้นมีซินโนนิม (Synonyms)

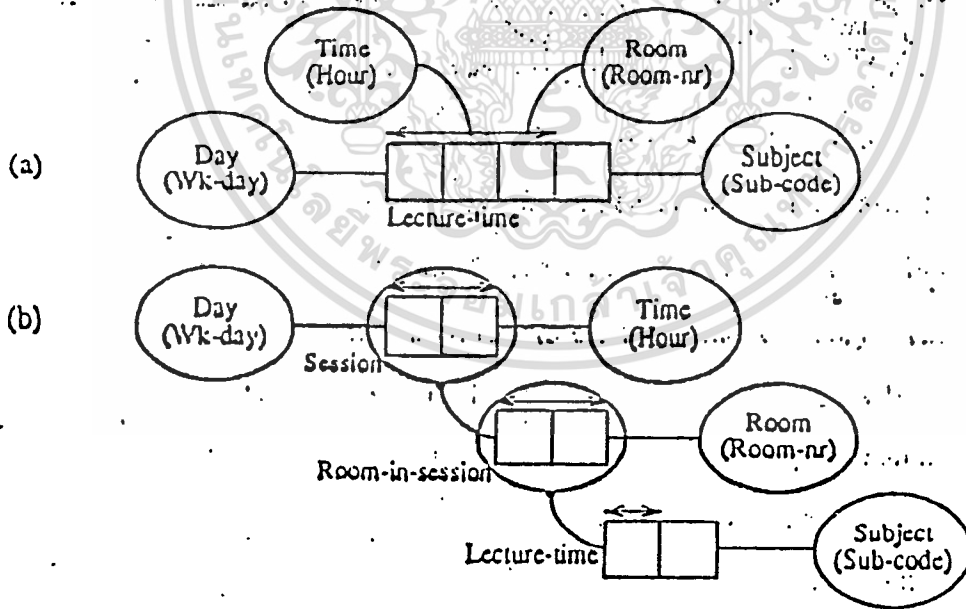
นิยามที่ 2 :

สิ่งที่ใช้อ้างถึงสิ่งที่เราสนใจแบบปรวมภูมิเป็นสิ่งที่ใช้อ้างถึงเพียงอย่างเดียวถ้าไม่มีซินโนนิม แต่ถ้ามีซินโนนิมจะถือว่าสิ่งที่เกิดขึ้นก่อนในนิยามที่ 1 เป็นสิ่งที่ใช้อ้างถึงสิ่งที่เราสนใจแบบปรวมภูมิ

3. คุณสมบัติบางประการของแบบแผนความคิดความเข้าใจรวมยอดของวิธีวิเคราะห์ข้อมูลของนิลเซน

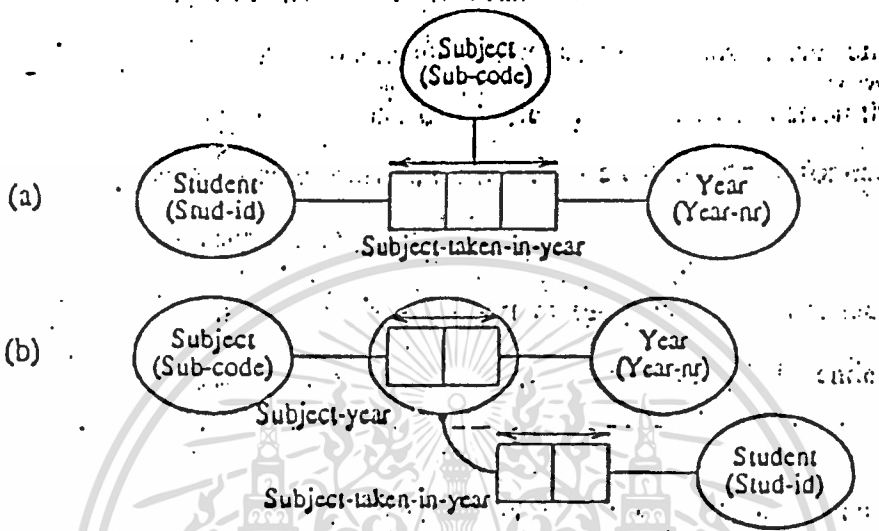
1) แต่ละแพลตฟอร์มที่มีโวลจจำนวน n (n -ary, $n \geq 2$) จะมีข้อกำหนดความเป็นหนึ่งนิยามบนแพลตฟอร์มในอย่างน้อยหนึ่งโวล

2) แต่ละแพลตฟอร์มที่มีโวลจจำนวน n (n -ary, $n \geq 2$) การมีข้อกำหนดความเป็นหนึ่งครอบคลุมนโวลจจำนวน $n - 1$ จะมีความหมายเหมือนกับแพลตฟอร์มแบบไบนารีที่มีร่างแหไปเกี่ยวข้องกับโวลของสิ่งอื่นแบบร่างแหในระดับที่ $n - 2$ ดังตัวอย่างที่แสดงในรูปที่ 2.15



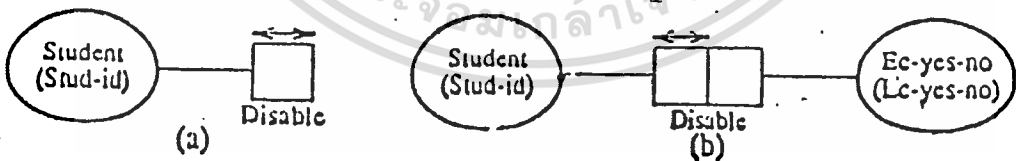
รูปที่ 2.15 ตัวอย่างคุณสมบัติข้อสอง : แบบแผน (a) เหมือนกับแบบแผน (b)

3) แต่ละแฟลส์ที่มีโรลจำนวน n และมีข้อกำหนดความเป็นหนึ่งครอนคลุมทุกโรลของแฟลส์จะมีความหมายเหมือนกับแฟลส์แบบไบนารีที่มีร่างแหไปเกี่ยวข้องกับโรลของสิ่งที่เราสนใจอันอื่น โดยมีข้อกำหนดแมนเดทอว์ปรากฏอยู่บนร่างแห ผลที่ได้จะเป็นร่างแหอันใหม่ออกมา ซึ่งจะไปเกี่ยวข้องกับโรลของสิ่งที่เราสนใจอันอื่นต่อๆ ไปอีก ดังแสดงในรูปที่ 2.16



รูปที่ 2.16 ตัวอย่างคุณสมบัติข้อ 3 : แบบแผน (a) เหมือนกับแบบแผน (b)

4) แฟลส์ที่มีโรลเดียว (Unary) และมีข้อกำหนดความเป็นหนึ่งครอนอยู่บนโรลนั้นมีความหมายเหมือนกับแฟลส์ที่มีโรลเดิมไปเกี่ยวข้องกับอีกโรลของสิ่งอันที่พิเศษ โดยมีข้อกำหนดแมนเดทอว์ และข้อกำหนดความเป็นหนึ่งปรากฏอยู่บนโรลเดิม ดังแสดงในรูปที่ 2.17



รูปที่ 2.17 ตัวอย่างคุณสมบัติข้อ 4 : แบบแผน (a) เหมือนกับแบบแผน (b)

4. ความหมายของแฟลส์แบบต่างๆ

ทฤษฎีบทที่ 1

แฟลส์ที่มีโรลจำนวน n (n -ary, $n \geq 2$) และมีข้อกำหนดความเป็นหนึ่งครอน

กลุ่มทุกโวลจะแทนผลคูณเวกเตอร์ (Multivalued Dependent ; MVD) ของสิ่งที่เราสนใจ
ทฤษฎีบทที่ 2

แพลตฟอร์มที่โวลจำนวน n ($n \geq 2$) และมีข้อกำหนดความเป็นหนึ่งรอบ
กลุ่มโวลจำนวน $n - 1$ จะแทนฟังก์ชันเวกเตอร์ (Functional Dependent ; FD)

5. แบบแผนรูปแบบปกติที่เหมาะสมที่สุด (Optimal Normal Form ; ONF Schema)

เราจะกล่าวว่าแบบแผนระบบฐานข้อมูลแบบสัมพันธ์ (Relational Database Schema) จะเป็นแบบแผนปกติที่เหมาะสมที่สุดก็ต่อเมื่อแบบแผนนั้นไม่มีการซ้ำซ้อนข้อมูล และมีจำนวนความสัมพันธ์น้อยที่สุด

การตรวจสอบอาร์ตี (ARITY CHECKING)

คือการตรวจสอบจำนวนอาร์ตี (จำนวนโวล) ของแพลตฟอร์มว่ามีมากน้อยหรือไม่อย่างไรหาก
มีมากเกินไปที่ควรจะเป็น เราอาจกล่าวว่าแพลตฟอร์มนั้นยาวเกินไป (too long) หากมีน้อยกว่าที่ควรเรา
กล่าวว่าแพลตฟอร์มนั้นสั้นเกินไป (too short) แพลตฟอร์มที่ยาวไปจะเป็นแพลตฟอร์มที่คอมพาว และ
แพลตฟอร์มที่สั้นไปก็จะเกิดการสูญเสียข้อมูลที่เราจะมีได้ ดังนั้นเราจะต้องทำให้จำนวนอาร์ตีถูกต้อง
เหมาะสม (right arity) (หนังสืออ้างอิง 12) หลักพื้นฐานของการตรวจสอบอาร์ตี คือหาก
ตรวจสอบดูแล้วสามารถแบ่งแยก (split) แพลตฟอร์มได้ก็ให้ทำการแบ่งจนกว่าจะแยกต่ออีกไม่ได้
การตรวจสอบมีอยู่ด้วยกัน 3 วิธี ที่จะใช้ร่วมกันคือ

1) ใช้สามัญสำนึกในการพิจารณาว่า field ต่างชนิดใด "ขึ้นกับ" (hook on) ชนิดใด
ชนิดที่ไม่ขึ้นแก่กัน เราสามารถทำการแบ่งแยกได้ ประเด็นสำคัญของวิธีนี้อยู่ที่ หากการแบ่งแยกแพลตฟอร์ม
ไหนปรากฏผลออกมาว่ามีการสูญหายของข้อมูล เราก็จะไม่ทำการแบ่งแยก

2) วิธีคีย์ที่สั้นที่สุด (Shortest Key) คือ ถ้าหากความยาวของแพลตฟอร์มมากกว่าความยาว
ของคีย์ที่สั้นที่สุดในแพลตฟอร์มในอย่างน้อยที่สุด 2 ชนิดหรือ 2 คอลัมในแพลตฟอร์มดังกล่าวย่อมสามารถ
ทำการแบ่งแยกได้ แต่หากไม่เป็นตามข้อกำหนดข้างต้นก็จะทำการแบ่งแยกไม่ได้

3) โปรเจกชันและจอยน์ (Projections and Joins) วิธีการนี้ค่อนข้างซับซ้อนกว่า 2
วิธีแรก ในรายละเอียดจะไม่ขอกล่าวถึงในที่นี้ ทั้งนี้เพราะการใช้วิธีการนี้ผู้ทำการตรวจสอบจะต้อง
มีตัวอย่างข้อมูลในแต่ละชนิดจริงอยู่ในมือ เพราะวิธีการนี้ขึ้นกับการเปรียบเทียบข้อมูลทั้งเดิมใน
ชนิดของแต่ละ โวล กับข้อมูลที่มีการยุ่งเกี่ยวกับออเพอเรชันของพีชคณิตแบบสัมพันธ์ (RELATIONAL
ALGEBRA OPERATION) สองออเพอเรชันคือโปรเจกชันและจอยน์ ว่ามีการแตกต่างกันในตัว
ของข้อมูลหรือไม่ เป็นเกณฑ์ในการตัดสินใจแบ่งแยก โดยเนื้อที่วิธีการข้อ 1 และ 2 น่าจะเพียงพอ
ต่อการตรวจสอบแล้ว ผู้สนใจสามารถค้นคว้าเพิ่มเติมได้จากหนังสืออ้างอิงท้ายเล่มหนังสืออ้างอิง
12]

ออปติมอล นอร์มัล ฟอม (OPTIMAL NORMAL FORM)

จากแผ่นสไลด์จะทำการสร้างตารางของข้อมูล โดยพยายามให้อยู่ในรูปออปติมอล นอร์มัล ฟอม ประเด็นสำคัญที่มีการมุ่งทำให้โครงสร้างของข้อมูลอยู่ในรูป ออปติมอล นอร์มัล ฟอม คือ

1. ความต้องการให้ได้โครงสร้างฐานข้อมูลที่ง่าย
2. ความต้องการให้ได้โครงสร้างฐานข้อมูล ที่มีการเข้าช้อนน้อยที่สุด เพื่อหลีกเลี่ยงความผิดพลาดจากการแก้ไข (update anomalies)
3. มีจำนวนตารางน้อยที่สุด เพื่อลดการจัดการข้อมูลระหว่างไฟล์ (INTER-FILE ACCESS) อันเป็นการเพิ่มประสิทธิภาพในการประมวลผล

โครงสร้างข้อมูลซึ่งอยู่ในรูปออปติมอล นอร์มัล ฟอมจะสามารถรับประกันสิ่งต่างๆที่กล่าวไปนี้

ออปติมอล นอร์มัล ฟอม อัลกอริทึม (OPTIMAL NORMAL FORM ALGORITHM)

ขั้นตอนการทำให้โครงสร้างข้อมูลอยู่ในรูปออปติมอล นอร์มัล ฟอม

1. สำหรับแต่ละแผ่นสไลด์ไทป์ที่ไม่มียูนิคีย์ (ไม่มีข้อกำหนดของความเป็นหนึ่งครอบคลุมอยู่บนโรลหนึ่งโรลใดเพียงหนึ่งโรล) ให้แยกออกมาสร้างตาราง
2. สำหรับเอ็นติตี้ไทป์ที่มียูนิคีย์หลายตัวมาเชื่อมอยู่ (attached) ให้ทำการรวมแผ่นสไลด์ไทป์ที่เกี่ยวข้องกันเอ็นติตี้ไทป์ดังกล่าวเข้าเป็นตารางๆเดียว
3. สำหรับแต่ละแผ่นสไลด์ไทป์ที่เหลือ (เอ็นติตี้ไทป์ที่มียูนิคีย์) ทำการสร้างตาราง
4. กำจัดการเข้าช้อนที่อาจเกิดขึ้นได้ จากการรวม 1:1 โรลไปยังตาราง 2 ตารางอันเกิดจากรัฐในข้อ 2 ซึ่งกำจัดการตัดผลลัพธ์ในทิ้งไปจากตารางหนึ่ง คงเหลือไว้แค่ตารางเดียวเท่านั้น

บทที่ 3
การสร้างโปรแกรม

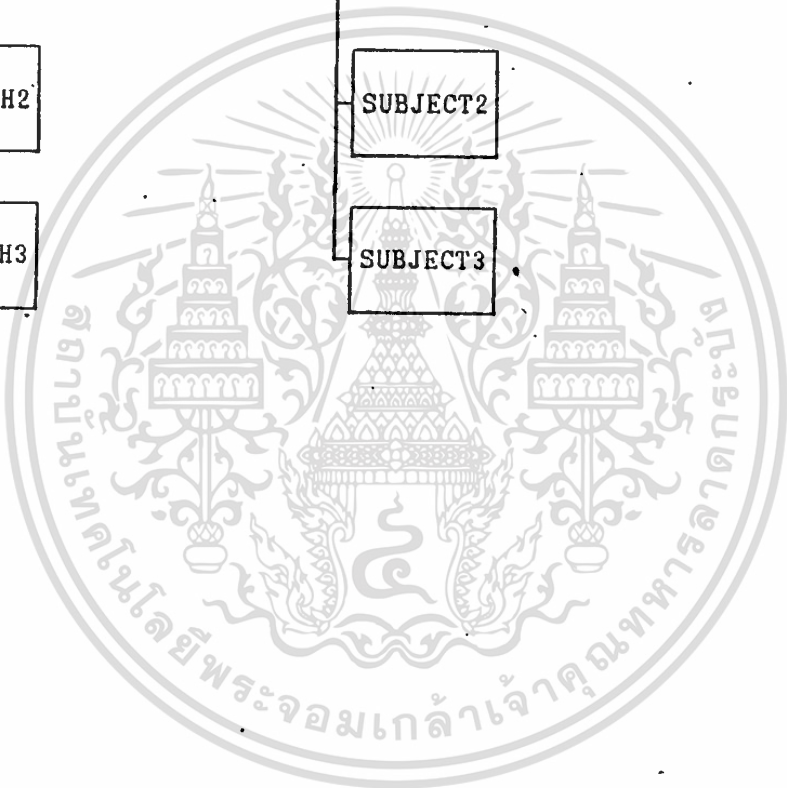
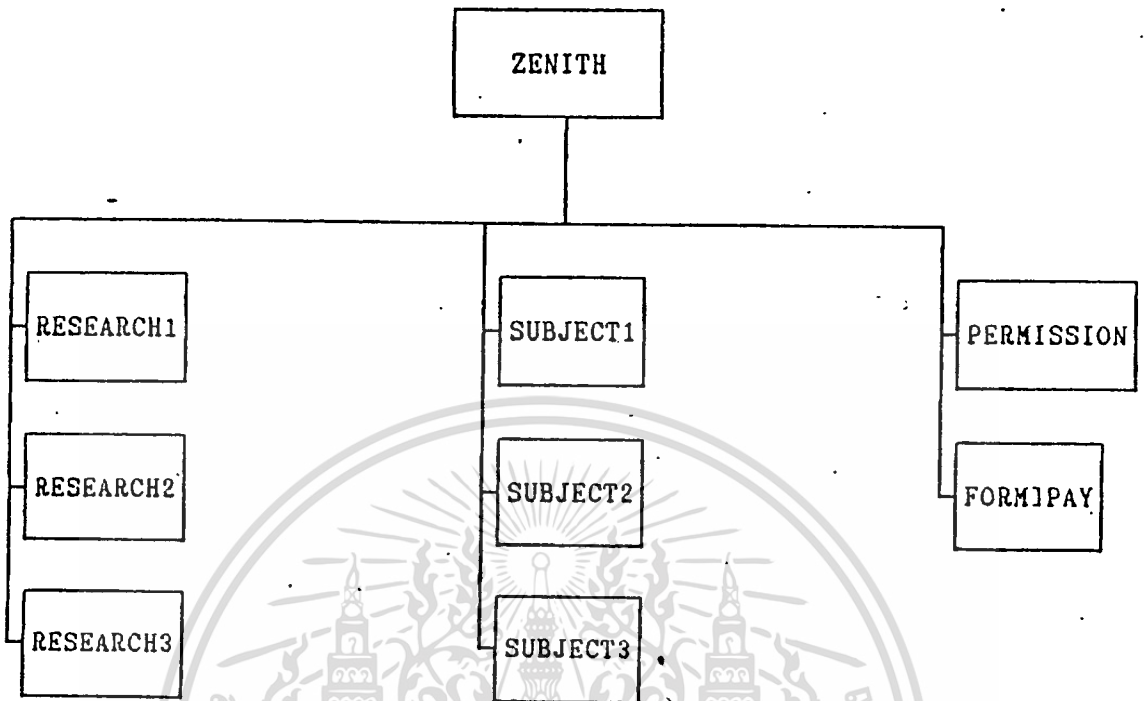
โปรแกรมระบบงานคอมพิวเตอร์นี้ ประกอบด้วยไฟล์หลัก 9 ไฟล์คือ

1. ZENITH ส่วน login ตรวจสอบ password และเป็นทางผ่านไปสู่ไฟล์อื่น
2. RESEARCH1 รายละเอียดงานวิจัย และการเบิกงบวิจัย
3. RESEARCH2 สรุปรายชื่องานวิจัย
4. RESEARCH3 สรุปการเบิกงบวิจัย
5. SUBJECT1 งานสอน
6. SUBJECT2 การคิดเกรด
7. SUBJECT3 สรุปรายชื่อวิชาที่สอน
8. PERMISSION ใบขออนุมัติข้อวัสดุ ใบเบิกวัสดุ
9. FORM1PAY ใบสำคัญจ่าย

ส่วนไฟล์ที่เหลือใช้เป็นส่วนประกอบย่อยคือ

1. prt.c ใช้ในการจัดการในการพิมพ์ข้อมูลออกทางเครื่องพิมพ์
2. uti.c ใช้ในการแสดงผลออกทางหน้าจอ
3. dtime.o ใช้ในการอ่านค่าเวลาจากระบบ unix

หลักการในการนำไฟล์ต่างๆมาทำงานร่วมกันได้ก็โดยการใช้วิธี function call จากไฟล์ zenith ซึ่งมีโครงสร้างดังนี้



ส่วนประกอบของโปรแกรม

VARIABLE DECLARATION

VERIFICATION

MENU FACILITY

INPUT SECTION

PRINTING

SPECIFIC MENU DETAIL

SQL CONNECTING

DATA MANIPULATION

ERROR DETECTION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบของโปรแกรม

1. Variable Declaration

เป็นส่วนที่ใช้ในการกำหนดค่าตัวแปร ที่จะใช้รับข้อมูล จากฐานข้อมูล โดยกำหนดเป็น varchar char และ int โดยตัวแปรแต่ละตัวจะต้องมีความสัมพันธ์กับขนาดของข้อมูล ที่เก็บในฐานข้อมูลและขนาดของการรับข้อมูลในส่วน Specific Menu Detail โดยส่วนนี้จะสังเกตได้จาก

```
EXEC SQL BEGIN DECLARE SECTION;
```

```
-----
```

```
EXEC SQL END DECLARE SECTION;
```

2. Verification

เป็นส่วนที่ใช้ในการรักษาความปลอดภัย (Security) ของระบบฐานข้อมูล นอกถึงสิทธิในการเข้าสู่โปรแกรมส่วนใด ๆ โดยอาศัยการผ่านค่า password

3. Menu Facility

ใช้ในการจัดการเกี่ยวกับการแสดงผลบนหน้าจอ ประกอบด้วยส่วนต่างๆ ดังนี้

- 3.1 การลบข้อความบนหน้าจอ ซึ่งสามารถกำหนด จุดเริ่มต้นและจุดสิ้นสุดของการลบได้
- 3.2 การนำข้อความไปแสดงผลบนหน้าจอ โดยกำหนดจุดเริ่มต้นและแอดทริบิวต์
- 3.3 การลบหน้าจอทั้งหมด ยกเว้นกรอบสี่เหลี่ยมของเมนู เมื่อมีการเปลี่ยนแปลงหน้าจอ
- 3.4 การตัดกรอบบนหน้าจอ แบ่งออกเป็นส่วนต่างๆ
- 3.5 การนำเวลามาจากระบบยูนิกซ์ แล้วเปลี่ยนให้อยู่ในรูปแบบที่ต้องการ

4. Input Section

ใช้ในการรับอักขระจากคีย์บอร์ด แล้วนำไปแสดงผลบนหน้าจอ โดยจะต้องรับตำแหน่งข้อมูล ความยาว ข้อมูลชุดเดิม และส่วนที่บอกว่าเป็นตัวอักษรหรือตัวเลข นอกจากนี้ยังมีการตรวจสอบ ฝั่งที่ขึ้นคีย์ต่างๆ เนื้ออ่านความสะดวกในการรับข้อมูล แล้วเก็บข้อความต่างๆ ใส่ตัวแปรแล้วส่งกลับมาสู่โปรแกรมหลัก

5. Printing

ใช้ในการพิมพ์ข้อความต่างๆ ออกทางเครื่องพิมพ์ โดยใช้ #include "prt.c" ซึ่งสามารถตรวจสอบโค้ดของเครื่องพิมพ์ที่ต่อกับคอมพิวเตอร์เครื่องนั้นอยู่ การทำงานของส่วนนี้จะส่งข้อความที่ต้องการพิมพ์ออกไปทีละบรรทัด โดยกำหนดตำแหน่งเริ่มต้นของการพิมพ์ ชนิดของข้อความ และการจัดตำแหน่งของข้อความ

6. Specific Menu Detail

นอกจากรายละเอียดในแต่ละฟิลด์ของข้อมูลว่ามีอะไรบ้าง ซึ่งรายละเอียดเหล่านี้ ใช้ในการแสดงผลบนหน้าจอ เพื่อให้ง่ายต่อการรับข้อมูลหรือแสดงผล โดยจะบอกถึงความหมายในแต่ละบรรทัดของข้อมูล ตัวแปรที่เก็บค่าแห่งการรับข้อมูล และความยาวของข้อมูลแต่ละฟิลด์

7. SQL Connecting

ใช้ในการเริ่มติดต่อกับฐานข้อมูลที่กำหนดไว้แล้ว โดยโปรแกรมที่จะสามารถเรียกใช้ฐานข้อมูลได้ จะต้องมียอดรหัส (IDCODE) และรหัสผ่าน (PASSWORD) ที่สามารถใช้กับฐานข้อมูลนั้นๆ ซึ่งตัวแปรเหล่านี้ System Administrator จะเป็นผู้กำหนดให้ โดยในโปรแกรมจะมีลักษณะดังนี้

```
EXEC SQL CONNECT :IDCODE IDENTIFIED BY :PASSWORD;
```

8. Data Manipulation

เป็นโครงสร้างที่ใช้ในการจัดการรับส่งข้อมูล ประกอบด้วย การค้นหา (Search) การเพิ่มเติม (Insert) การแก้ไข (Update) การลบ (Delete) การพิมพ์ (Print)

การค้นหา เป็นค่าแรกที่จะนำไปสู่ การแก้ไข การลบ และการพิมพ์ ซึ่งการค้นหาในโปรแกรมนี้จะมี 2 ลักษณะคือ ค้นหาข้อมูลที่เจาะจงเลข หรืออีกแบบหนึ่งก็คือ ค้นหาข้อมูลทั้งหมดมา แล้วเลือกดูข้อมูลที่ต้องการบนหน้าจออีกที ถ้าเกิดข้อผิดพลาด ในการหาข้อมูลไม่พบ ก็จะมีข้อความแจ้งข้อผิดพลาดออกมา (Error Message)

การเพิ่มเติม ใช้งานร่วมกับส่วน Input Section รับข้อมูลแต่ละฟิลด์มาเก็บไว้ในตัวแปร และแสดงออกมาบนหน้าจอ จากนั้นก็จะนำข้อมูลที่รับเข้ามาทุกฟิลด์ไปเก็บไว้ในตารางของฐานข้อมูลที่กำหนดไว้

การแก้ไข เมื่อค้นหาข้อมูลมาเรียบร้อยแล้ว ก็จะใช้คีย์(Keyword) ของข้อมูลที่ค้นหาหามาใช้ในการแก้ไข โดยจะต้องใช้ส่วน Input Section ในการเปลี่ยนแปลงแก้ไขข้อมูลในฟิลด์ที่ต้องการ แล้วจึงนำคีย์นั้นมาแก้ไข(Update) ในตารางของฐานข้อมูล

การลบ ก็ใช้คีย์ของข้อมูลที่ค้นหาหามาใช้ในการลบข้อมูลที่มีคีย์นั้นอยู่ทั้งหมดในฐานข้อมูล ซึ่งอาจจะประกอบด้วยหลายตาราง นอกจากนั้นก็ลบตามลักษณะชิ้นงาน ซึ่งอาจจะแค่ลบบางฟิลด์ หรือลบข้อมูลหมดทั้งชุด

การพิมพ์ เรียกว่าใช้โปรแกรมส่วน Printing ซึ่งจะจัดฟอร์มการส่งข้อมูลออกไปพิมพ์ทางเครื่องพิมพ์ที่ละบรรทัด โดยจะมีการส่งชนิดการพิมพ์ และส่งข้อมูลออกมานิพิมพ์ทางเครื่องพิมพ์ที่อยู่กับเทอร์มินัลตัวนั้นๆ

9. Error Detection

เมื่อเกิดข้อผิดพลาดใดๆ ในการติดต่อกับฐานข้อมูล ซึ่งจะมีลำดับของการผิดพลาด ถ้าเป็นการค้นหาข้อมูล หรือการจัดการข้อมูลอื่นๆแล้วมีข้อผิดพลาดก็จะมีข้อความแจ้งออกมาในช่องข่าวสาร ด้านล่างของเมนู แต่ถ้ามีข้อผิดพลาดร้ายแรงในการจัดการฐานข้อมูล ก็จะมีการยกเลิกการจัดการข้อมูลที่ผิดพลาดเหล่านั้น ที่เรียกว่า Rollback Work Release

บทที่ 4
ผลการวิเคราะห์

จากการศึกษาความเป็นไปได้ในการนำคอมพิวเตอร์มาใช้งาน ในภาควิชาวิศวกรรมคอมพิวเตอร์ จึงได้เลือกงานบางอย่างที่จะทดลองใช้คอมพิวเตอร์ คือ

- เอกสารหึ่งเบิก
- งานสอน
- งานวิจัย

จากไอแซค ซึ่งเป็นขั้นตอนที่จะพยายามทำความเข้าใจกับกิจกรรมต่างๆของภาควิชา ก่อนที่จะก้าวไปถึงขั้นการวิเคราะห์และพัฒนาระบบ นั่นคือก่อนที่จะวิเคราะห์ระบบได้นั้น จะต้องทราบถึง ระบบงานทั้งหมดที่กำลังดำเนินอยู่ เพื่อนำมาประกอบการพิจารณาหาทางแก้ไขปัญหาที่เกิดขึ้นกับระบบปัจจุบัน ก็จะได้เอ-กรานซึ่งจะสามารถวิเคราะห์ขั้นตอนการทำงาน ซึ่งจะเรียงตามลำดับ ตั้งแต่ระบบงานระดับสูง ไปจนถึงงานย่อยๆ ทั้งหมด

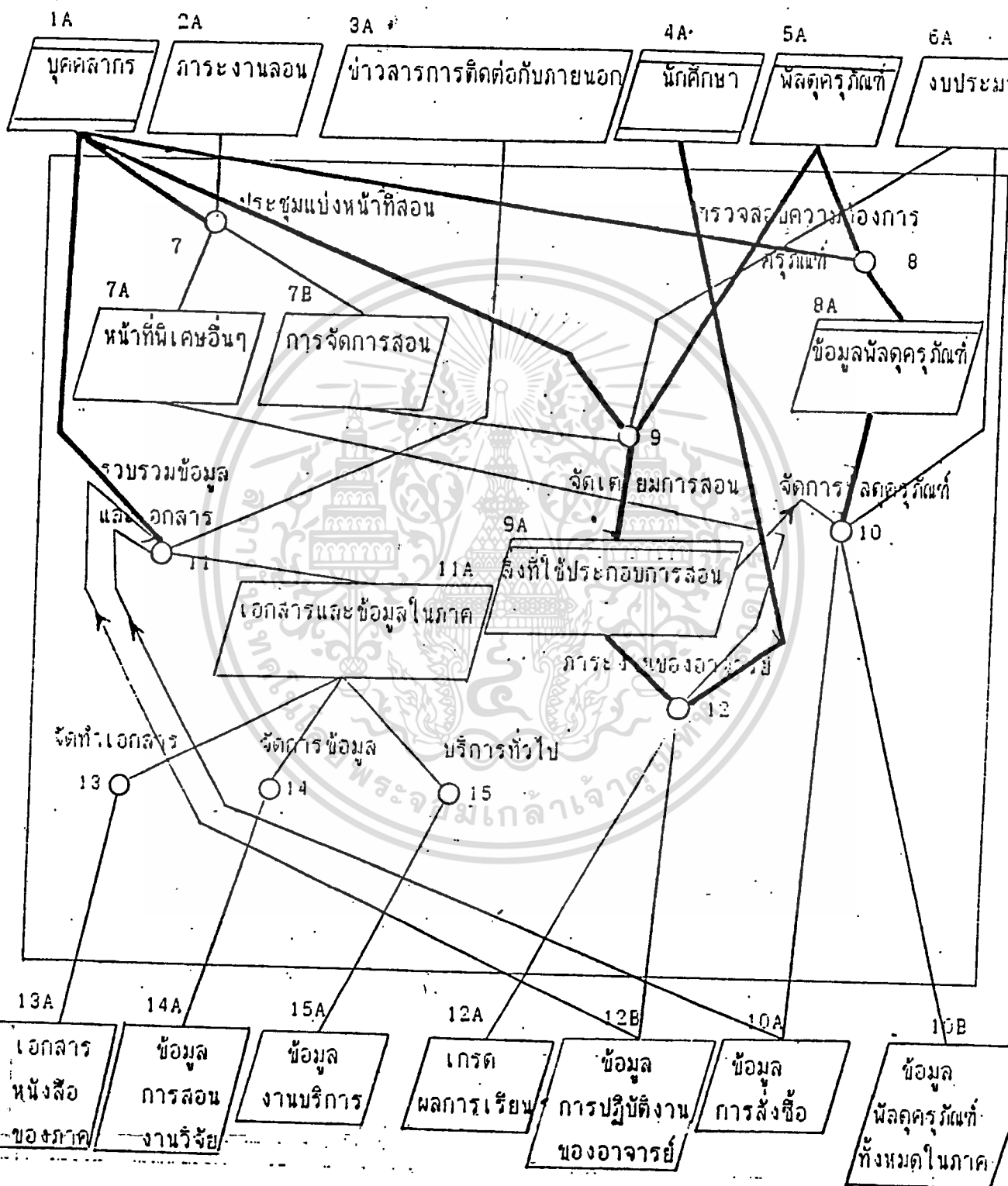
แล้วจึงได้ใช้ในแอมในการหาความสัมพันธ์ระหว่างข้อมูลออกมา สามารถเขียนเป็นแผนผังได้ดังรูป ซึ่งในแผนภาพในแอมจะประกอบด้วยแผลไทท และเอนดีตีไทท ที่จะสามารถครอบคลุมถึงข้อมูลทุกอย่างที่ต้องการ จากนั้นก็นำข้อมูลที่มีกลุ่มความสัมพันธ์อย่างเดียวกัน มารวมกันไว้ในตารางเดียวกัน แล้วจึงกระทำให้เกิด ONF (Optimal Normal Form) ก็จะได้ผลทั้งหมดดังนี้

Analyst: รัตนะ รัตนพันธ์
พงษ์สันต์ วชิรวงศ์ทอง

Date: 1989-08-29

No. CC

Subject: การบริหารงานของภาควิชาคอมพิวเตอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีสิทธิ์ที่คณะผู้บริหารและรองอธิการบดีแจ้งขอเอกสารทุกครั้งที่มีการนำไปใช้

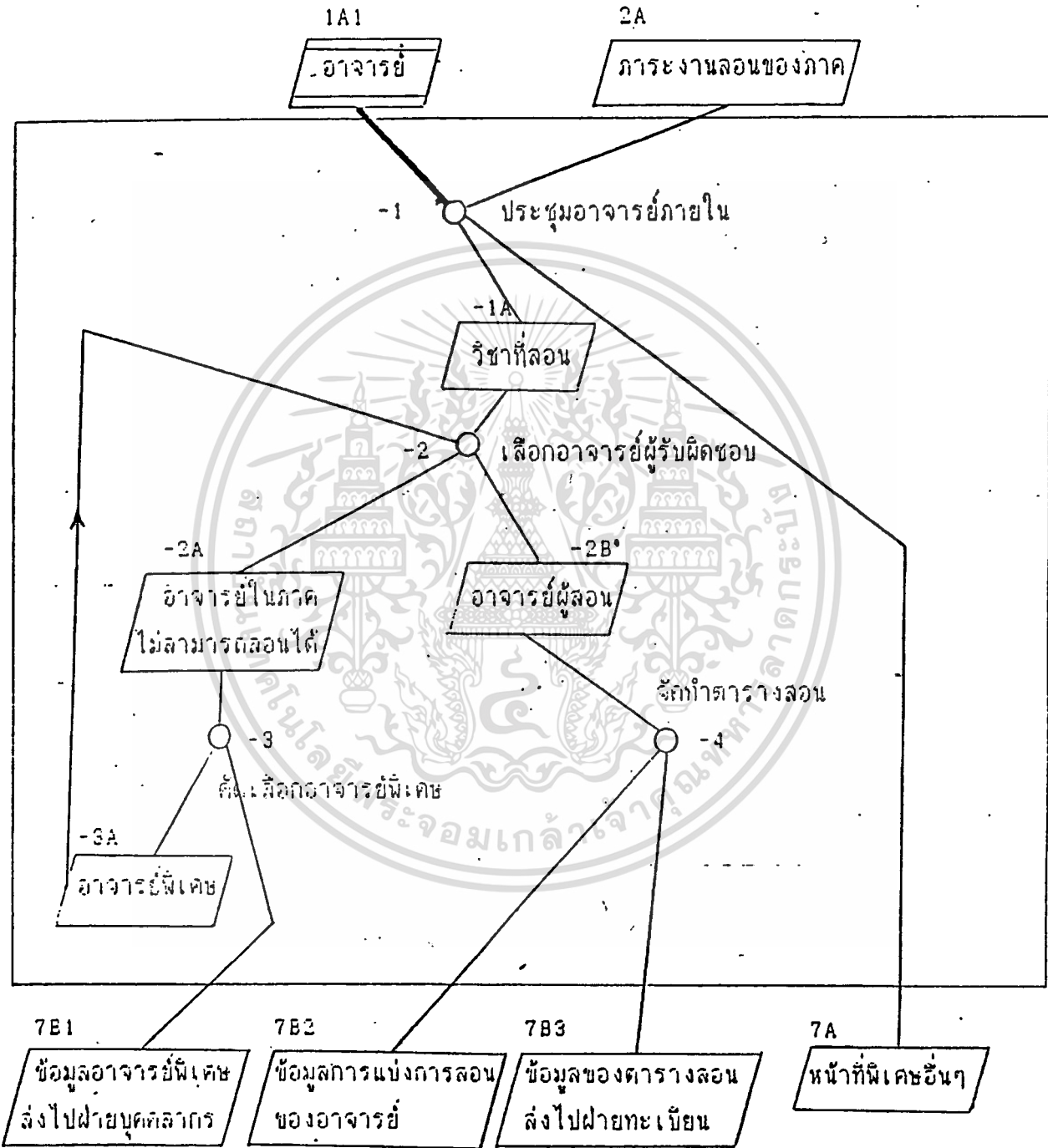
Analyst: รัตนะ รัตนพันธ์

พงษ์สันต์ วิชาวงศ์ทอง

Date: 1989-08-29

No. C7

Subject: ประชุมแบ่งหน้าที่สอน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ทำกรณีโดยทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

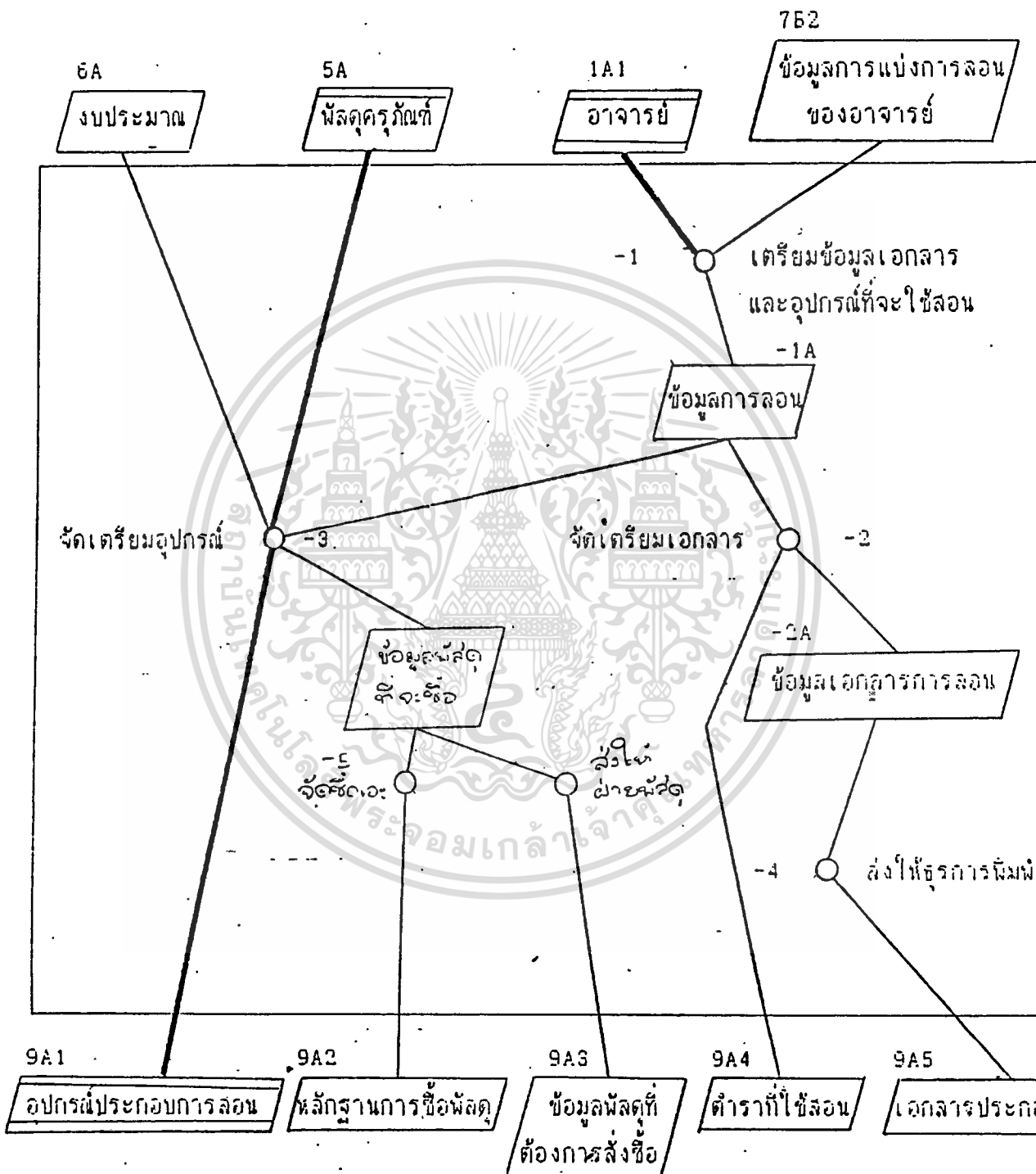
Analyst: รัตนะ รัตนพันธ์

พงษ์สันต์ วชิรวงศ์ทอง

Date: 1989-08-29

No. C

Subject: จัดเตรียมการสอน

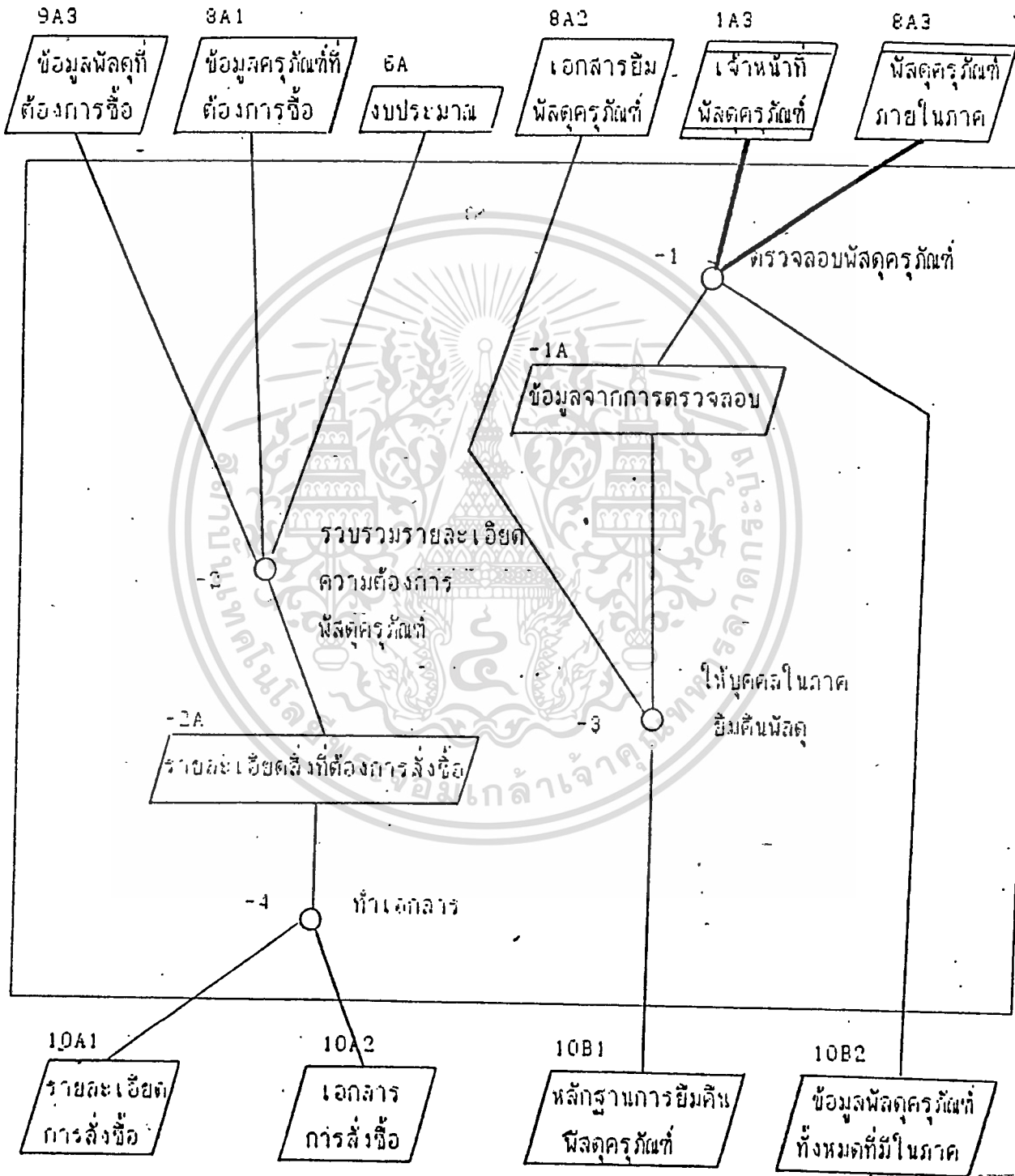


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ...
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเผยแพร่ และต้องอ้างอิถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Analyst: รัตนะ รัตนพันธ์
พงษ์สันต์ วัชรวงศ์ทอง
Subject: การจัดการพัสดุครุภัณฑ์

Date: 1989-08-29

No. 0

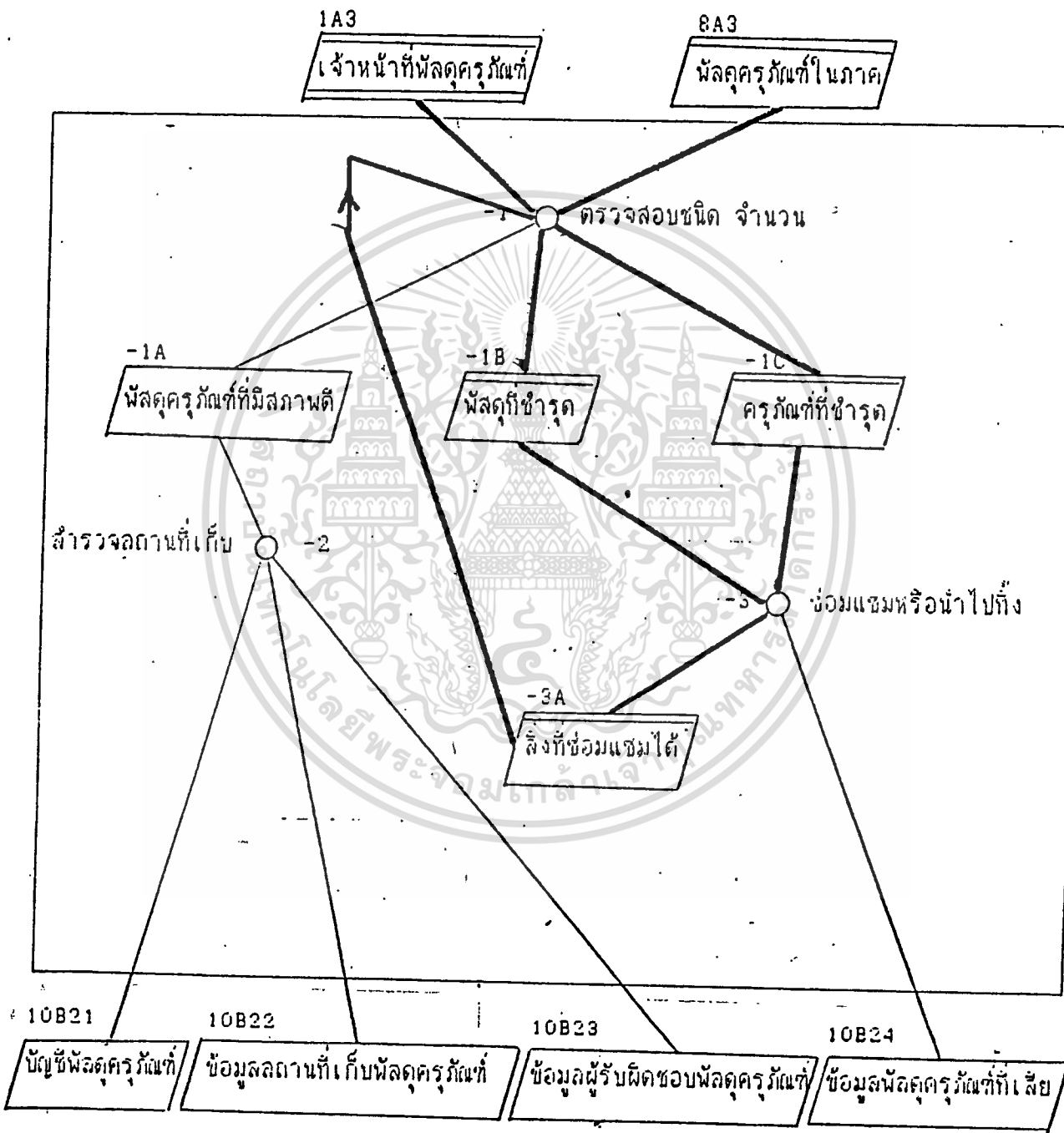


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแบบลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Analyst: รัตนะ รัตน์พันธ์
พงษ์สันต์ วชิรวงศ์ทอง
Subject: การตรวจสอบพัสดุครุภัณฑ์

Date: 1989-08-29

No. C



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

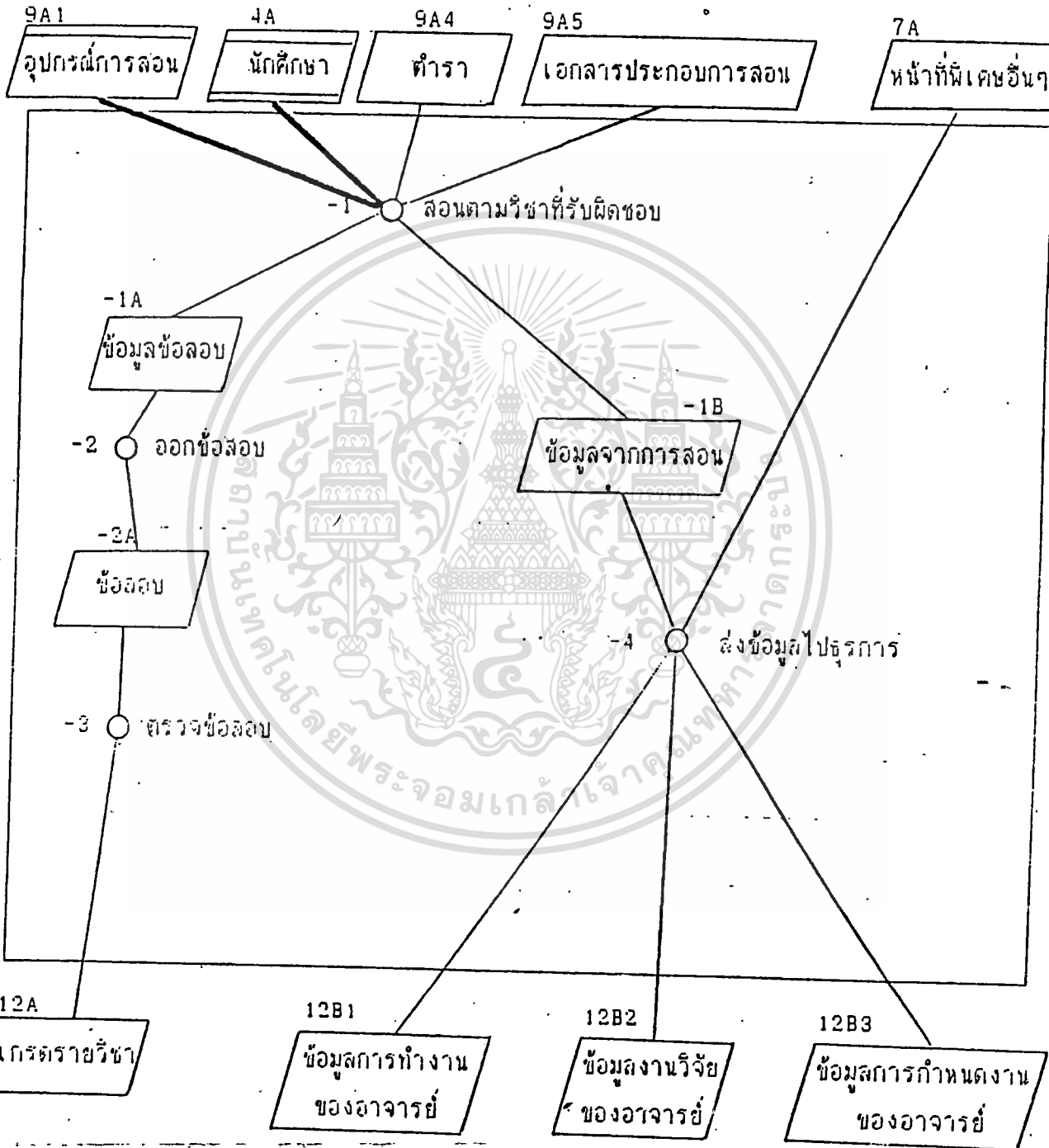
Analyst: รัตนะ รัตนพันธ์

พงษ์สันต์ วชิรวงศ์ทอง

Date: 1989-08-29

No. C1

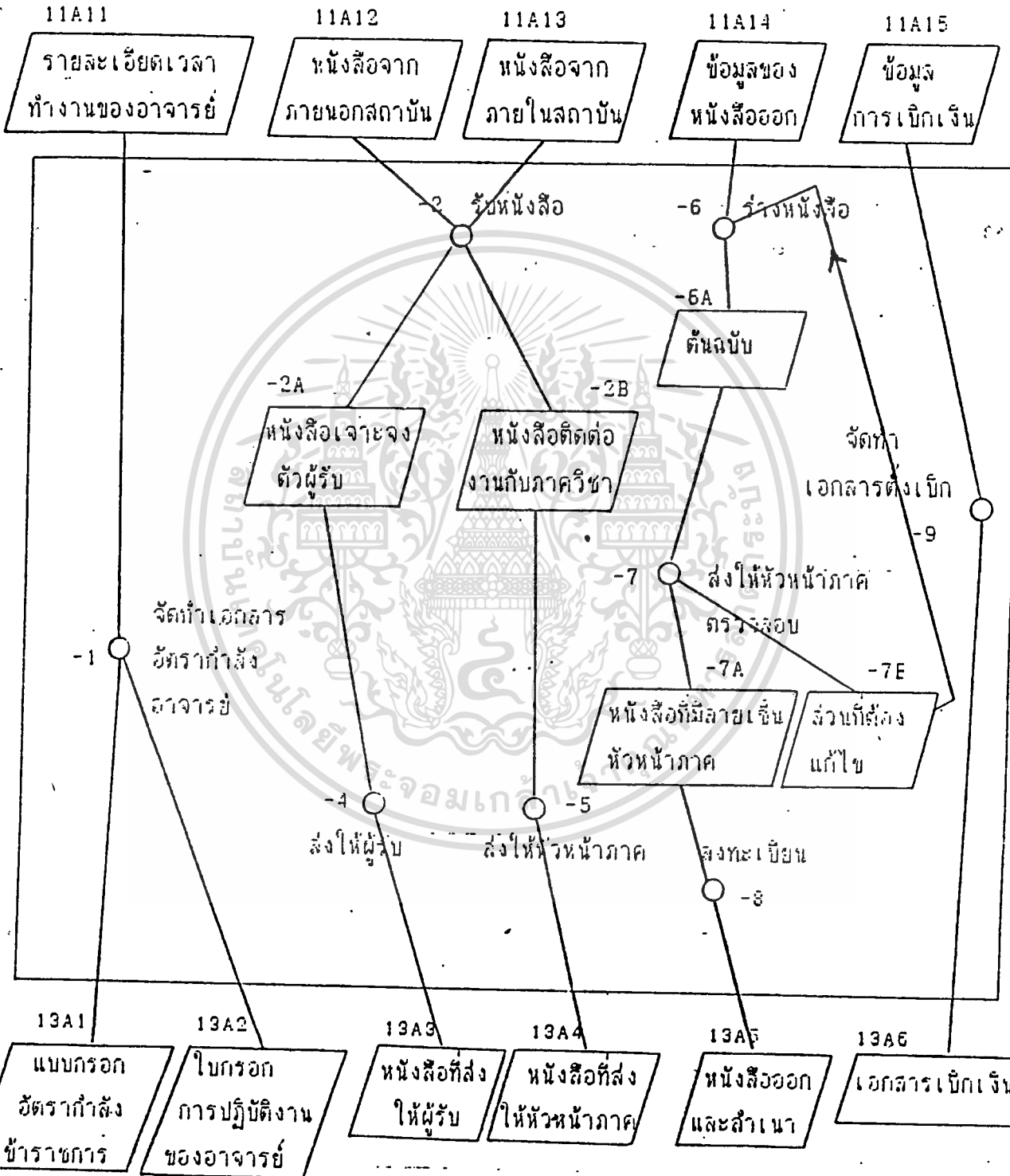
Subject: ภาระงานของอาจารย์



Analyst: รัตนะ รัตนพันธ์
พงษ์สันต์ วชิรวงศ์ทอง
Subject: จัดทำเอกสาร

Date: 1989-08-29

No. C



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Analyst: รัตนะ รัตนพันธ์
พงษ์สันต์ วชิรวงศ์ทอง
Subject: การเอกสารตั้งเบิก

Date: 1989-08-29

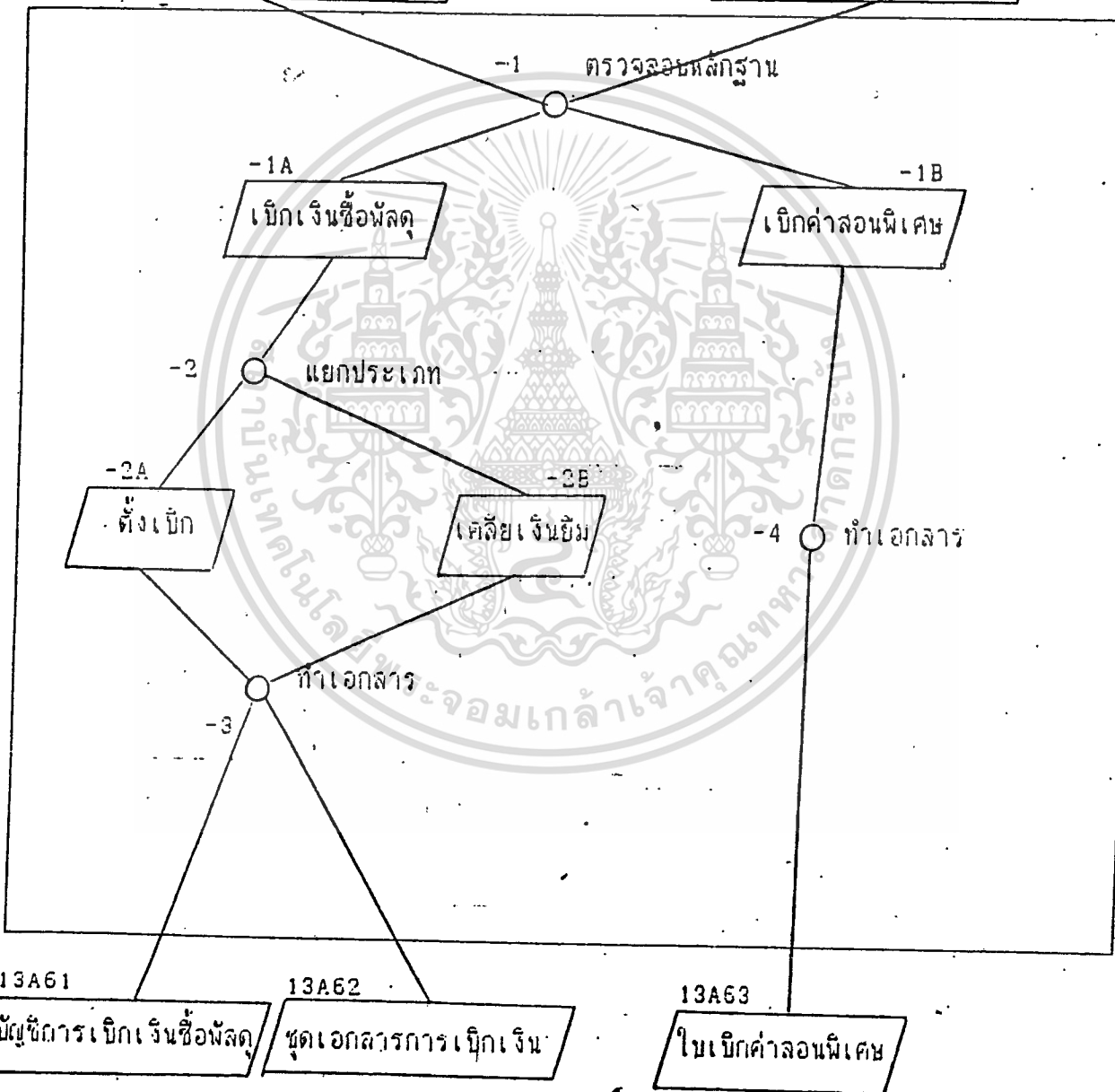
No. C1

11A141

หลักฐานการซื้อพัสดุจากอาจารย์

11A142

ข้อมูลการสอนของอาจารย์

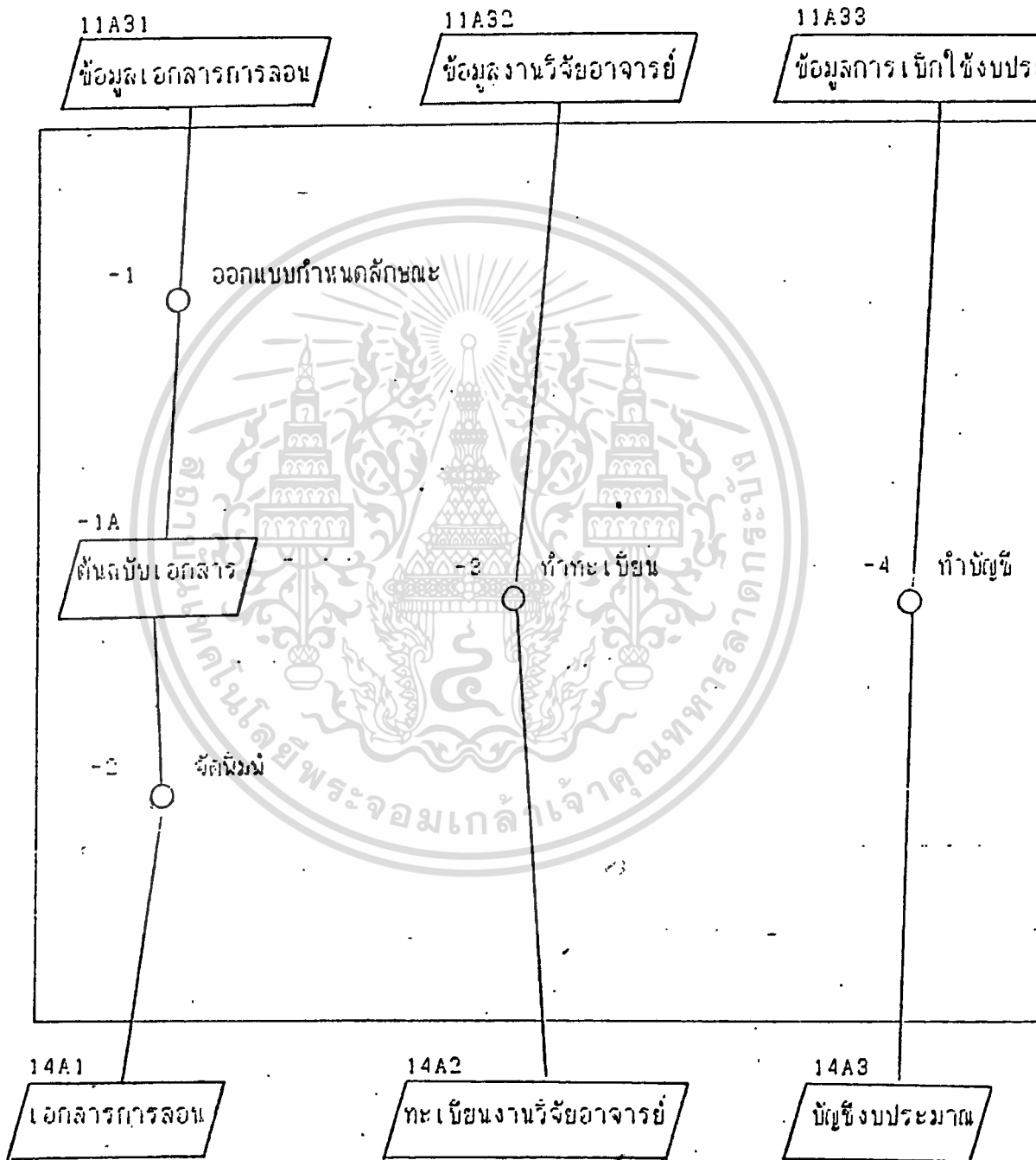


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Analyst: รัตนะ รัตนพันธ์
พงษ์สันต์ วัชรวิงค์ทอง

Date: 1989-08-29

Subject: การรวบรวมข้อมูล



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

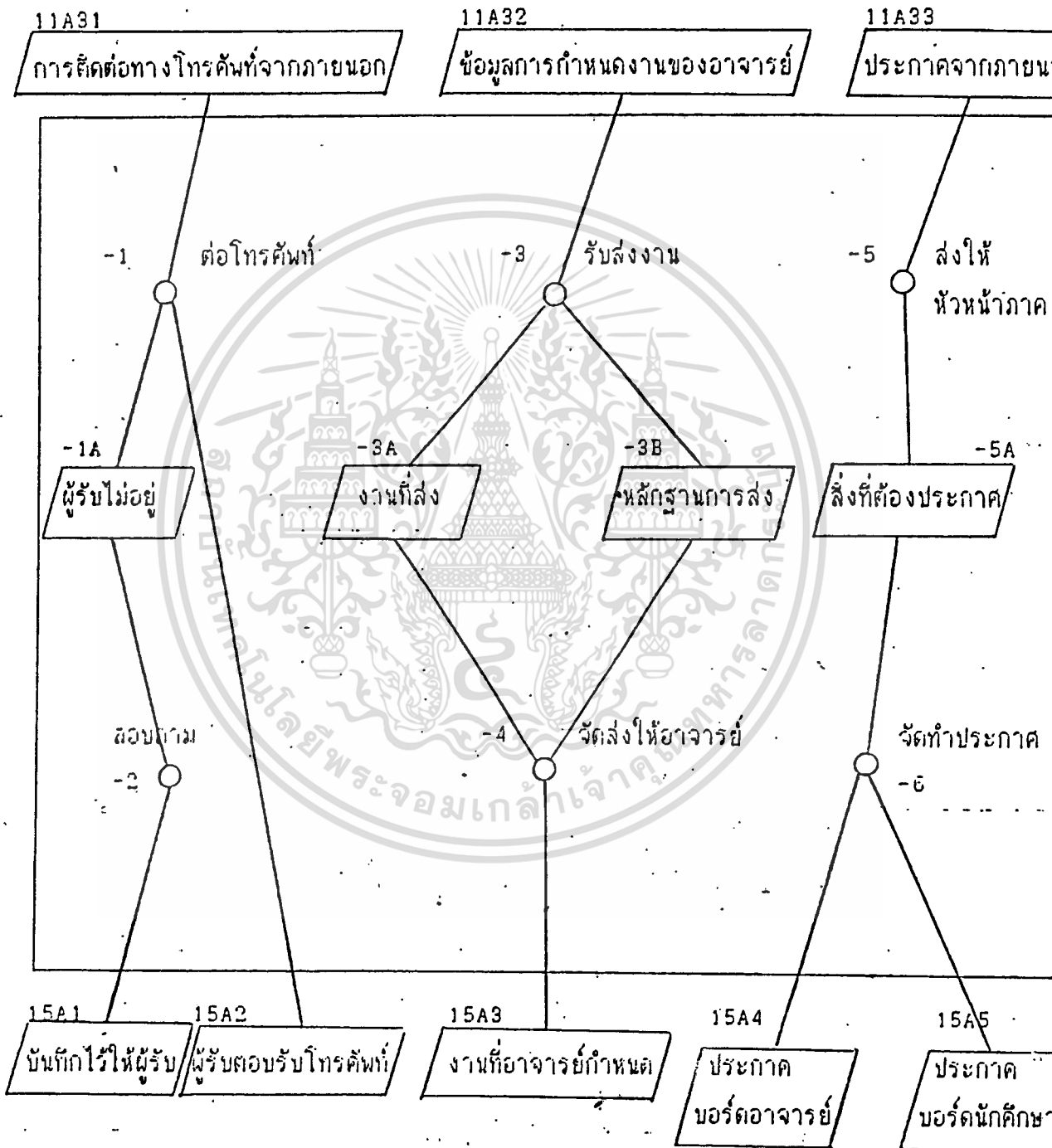
Analyst: รัตนะ รัตนพันธ์

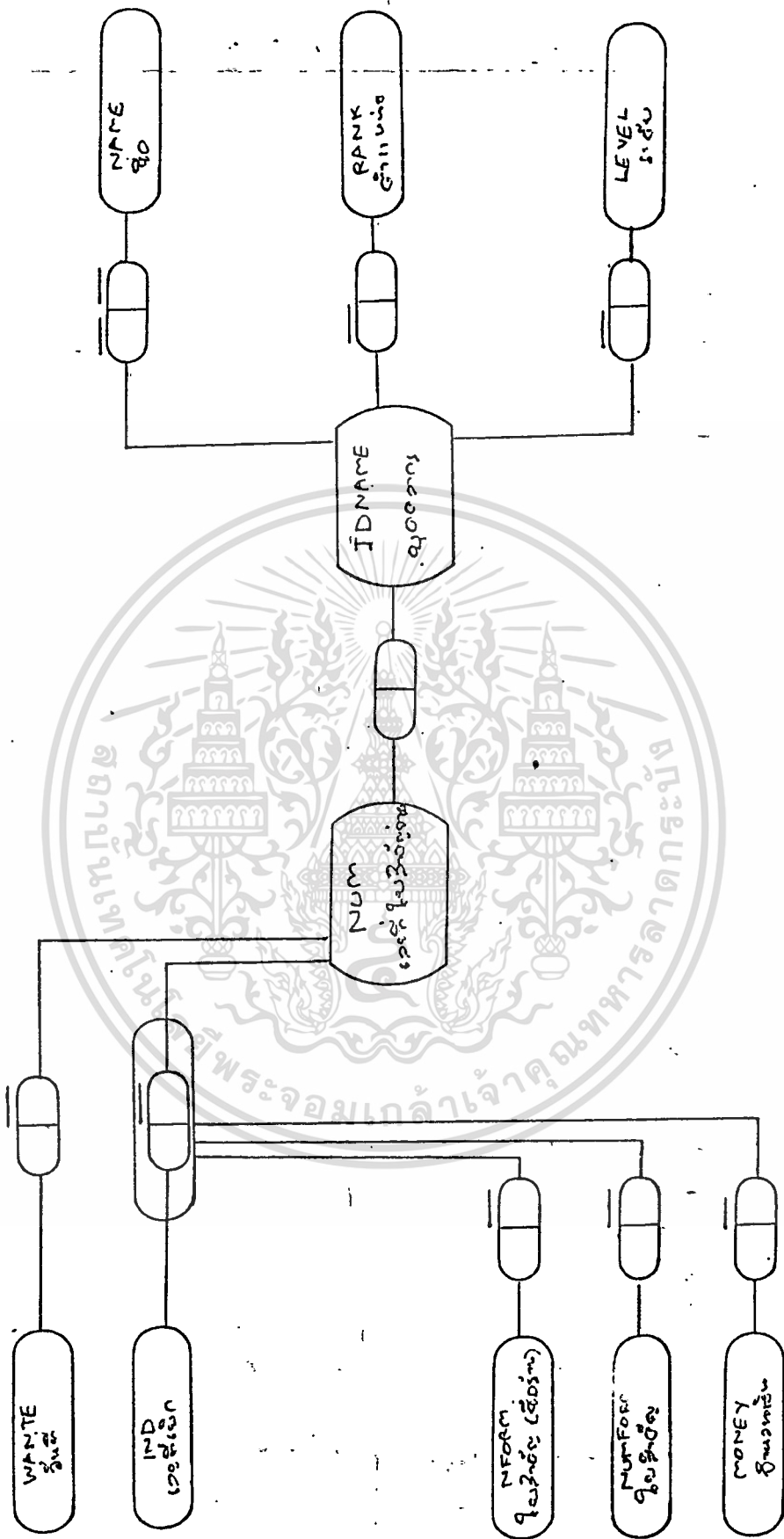
พงษ์สันต์ วัชรวงศ์ทอง

Date: 1989-08-29

No. C

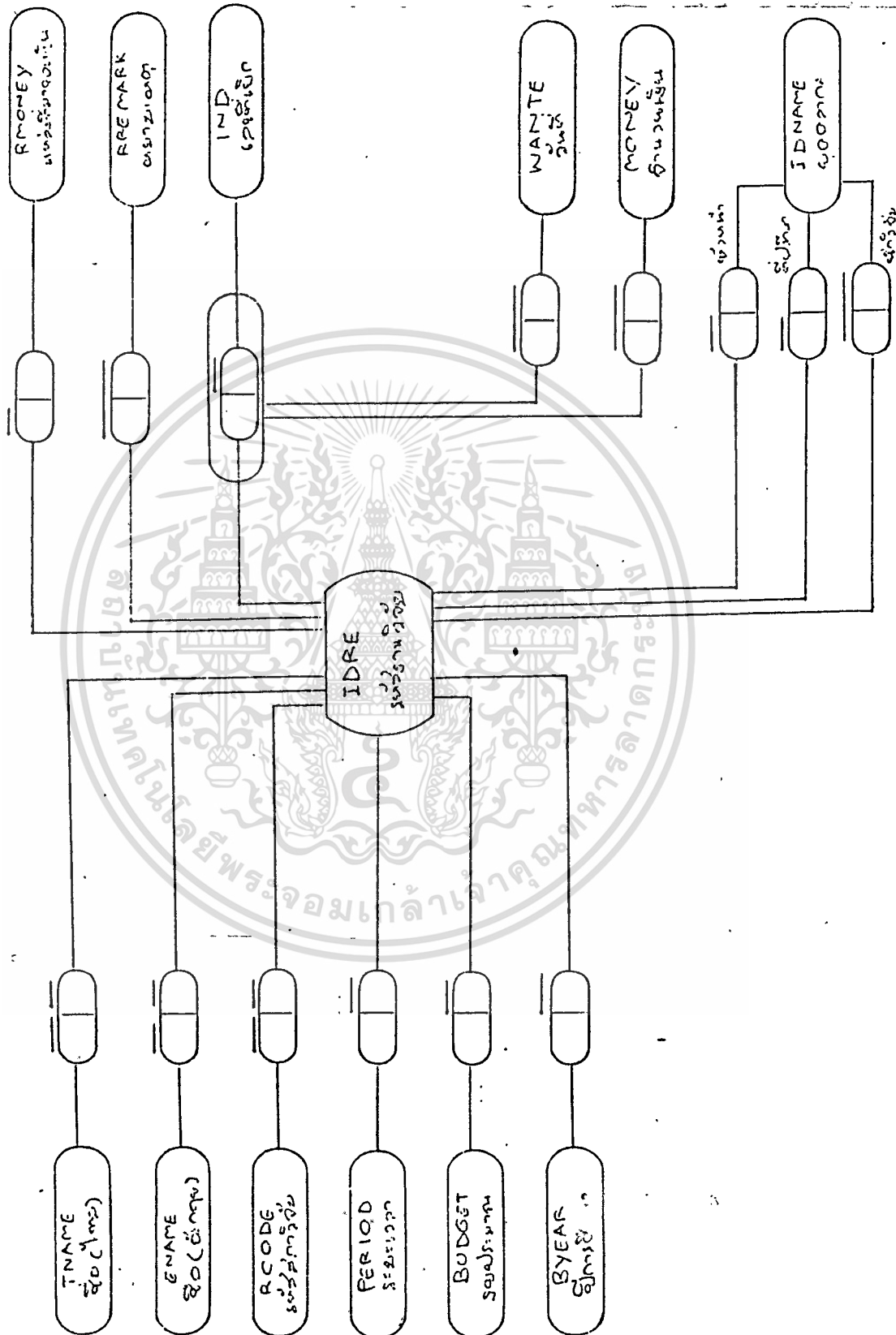
Subject: การจัดการด้านบริการทั่วไป



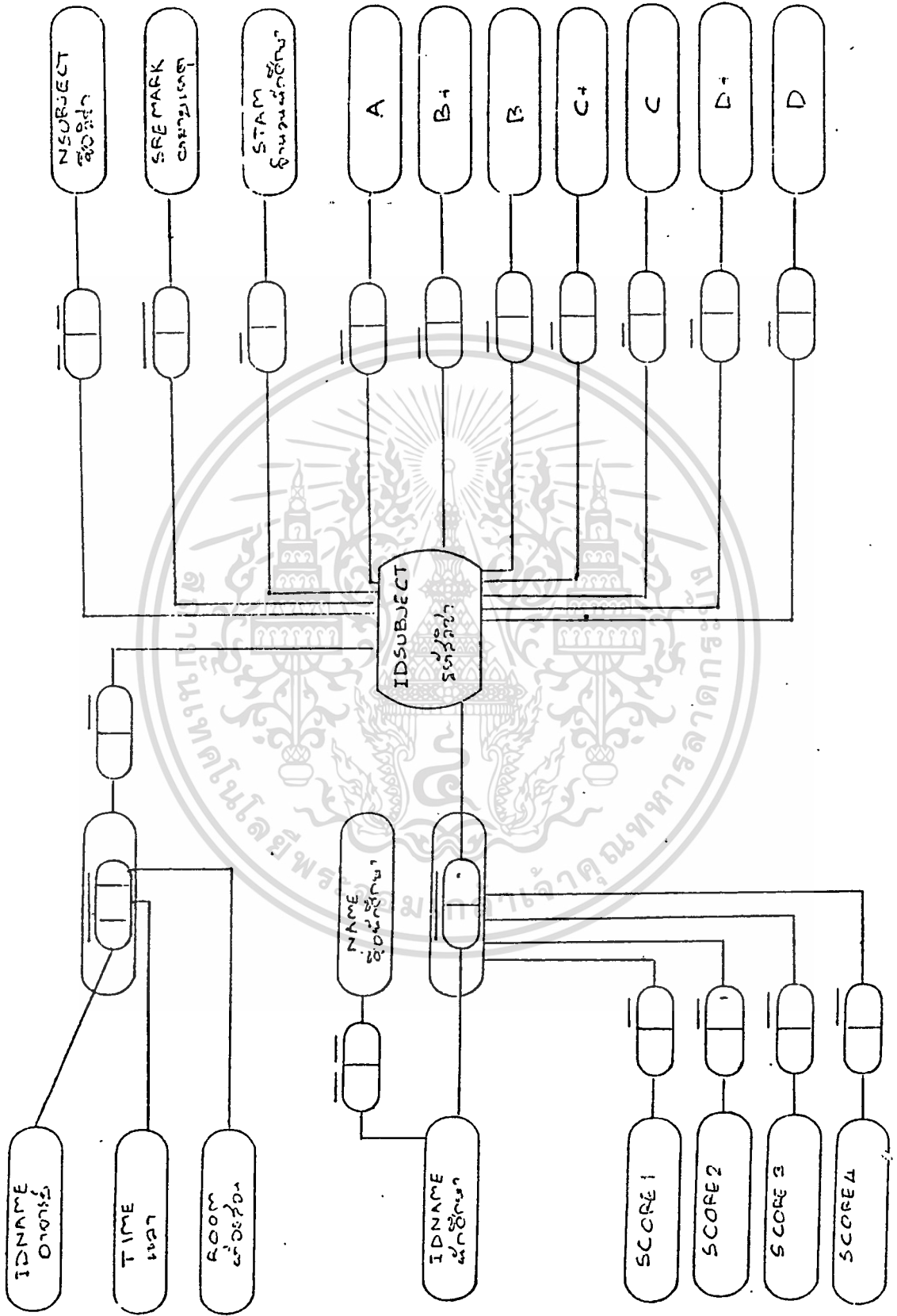


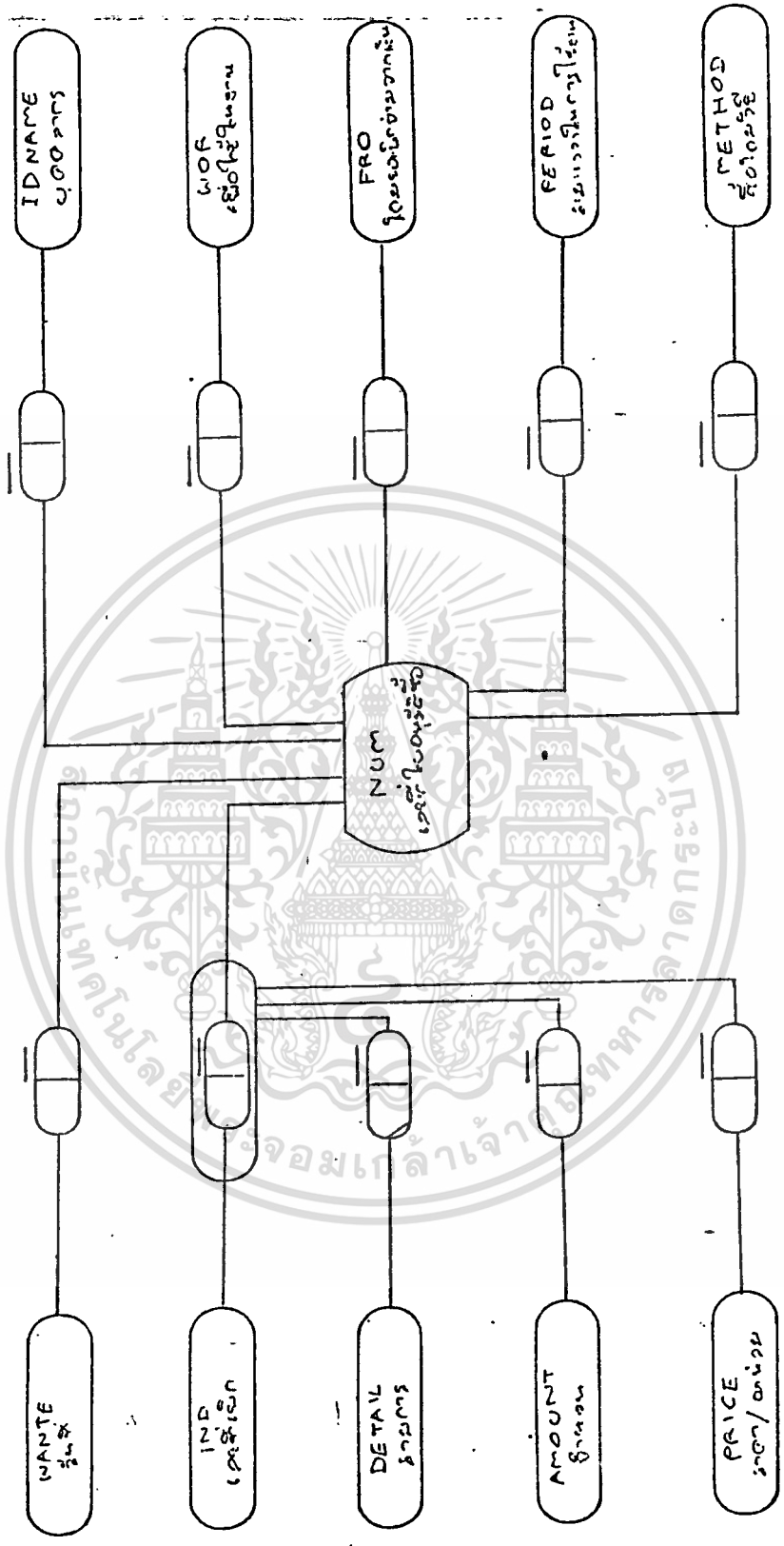
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และแจ้งถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





ปริศยานิพนธ์ชิ้นนี้เป็นการศึกษาข้อมูลเบื้องต้น ในการนำคอมพิวเตอร์มาประยุกต์ใช้ กับหน่วยงานระดับภาควิชา โดยศึกษาความเป็นไปได้และขอบเขตที่สามารถที่จะนำคอมพิวเตอร์มา ใช้ในงานได้ และได้พัฒนาโปรแกรมซึ่งเป็นต้นแบบของระบบที่จะพัฒนาต่อไปในอนาคต

ในหน่วยงานอย่างภาควิชาวิศวกรรมคอมพิวเตอร์ สมควรมีการใช้คอมพิวเตอร์ช่วย ในการบริหารงานเพื่อเป็นตัวอย่างแก่หน่วยงานอื่นๆ โดยจากการวิเคราะห์ปัญหาที่เกิดขึ้นจะพบว่า ไม่สามารถนำคอมพิวเตอร์มาใช้ได้ในงานทุก ๆ อย่างจึงได้เลือกใช้คอมพิวเตอร์ในงานที่เหมาะสม เท่านั้น ซึ่งก็คืองานที่มีขั้นตอนการทำงาน และมีข้อมูลที่ต้องการใช้อย่างชัดเจน

การนำคอมพิวเตอร์มาใช้ในการบริหารงานของภาควิชา จะเป็นการช่วยในการทำ งานต่าง ๆ ที่ปฏิบัติกันอยู่แล้ว ซึ่งจะช่วยให้อาจารย์และเจ้าหน้าที่ที่สามารถที่จะปฏิบัติงานได้สะดวก รวดเร็วยิ่งขึ้น ใช้คอมพิวเตอร์ในการทำงานที่มีขั้นตอนการทำงานซ้ำ ๆ กัน

การพัฒนาในขั้นต่อไป จะกระทำได้ในรูปของการขยายระบบข้อมูล ไปใช้ในงาน ด้านอื่น ๆ ที่สามารถทำได้โดยง่าย เนื่องจากใช้ฟังก์ชันการทำงานชุดเดียวกัน อย่างเช่นการ รับข้อมูลจากคีย์บอร์ด ขั้นตอนการทำงานของแต่ละหน้าจอก็คล้ายคลึงกันหมด แตกต่างกันก็เพียงแต่ รายละเอียดเล็กน้อยเท่านั้น

ภาคผนวก ก-

Rand Editor

Rand Editor เป็น editor ตัวหนึ่งบน UNIX ซึ่งมีความน่าใช้มาก เนื่องจากการทำงานที่ใกล้เคียงกับ Word processor หรือ editor ของตัวแปรภาษาที่ใช้กันแพร่หลายบน pc. ทำให้เกิดความคุ้นเคยและคล่องตัวในการใช้งาน ประกอบกับการใช้ที่เรียนรู้ได้ง่าย จึงสามารถที่จะใช้งานพื้นฐานในการสร้าง, แก้ไข file และสร้าง source file ของโปรแกรมภาษาต่าง ๆ ได้ในเวลาไม่มากนัก จึงเป็นประโยชน์อย่างยิ่งในการใช้งานบนระบบ UNIX เนื่องจาก vi editor ก็ใช้งานยาก นอกจากนั้นในการ transfer file จากระบบ dos มายังระบบ UNIX ก็สามารถที่จะใช้ Rand editor ลบ end of file (^Z) ใน file dos ได้แม้ว่าจะจะเป็น file ภาษาไทย ในขณะที่ vi editor ทำได้แต่ file ที่ไม่ได้ใช้ภาษาไทย

การใช้งาน Rand editor ปุ่มควบคุมและตำแหน่งของปุ่มควบคุม รวมทั้งคำสั่งต่าง ๆ อาจเปลี่ยนแปลงกันไปตามแต่การ setup terminal ของระบบ ในที่นี้จะใช้ pc tangon ของ NIXDORF เป็น terminal ของระบบ UNIX ที่ใช้ โดยบางปุ่มอาจจะแตกต่างกันไปบ้างจากคู่มือของระบบ UNIX

ในการอธิบายการใช้งานนี้จะอธิบายถึงคำสั่งและการทำงานที่จำเป็น และหมายเลขในการสร้างแก้ไข file ส่วนรายละเอียดอื่น ๆ นั้นสามารถศึกษาได้จากคู่มือ Rand editor

การเรียกใช้ Rand editor

การใช้งานจะเรียกใช้ Rand editor จาก shell command mode โดยใส่คำสั่ง e ตามด้วยชื่อ file ที่ต้องการสร้างหรือแก้ไข ดังนี้

```
e [ filename ]
```

ถ้า filename นั้นมีอยู่ใน Current directory อยู่แล้วก็จะเรียก file นั้นขึ้นมาปรากฏยังจอภาพ แต่ถ้ายังไม่มีการสร้าง file นั้น ก็จะถามว่าจะสร้าง file นั้นหรือไม่

```
Do you want to create <filename>. ?
```

ถ้าตอบ y ก็จะเข้าไปสร้าง file ใน window ถ้าตอบ n ก็จะ ออกไปยัง shell ถ้าเราเรียกใช้ Rand โดยไม่ได้ใช้ชื่อ file Rand ก็จะนำเอา file ล่าสุดที่เพิ่งเรียกใช้ขึ้นมาให้โดยอัตโนมัติ

เมื่อเรียกใช้ Rand และ file ถูกเรียกมาแล้วก็จะมาปรากฏที่หน้าจอโดยจะอยู่ในกรอบของ window มุมล่างขวาจะบอกตำแหน่งและชื่อ file มุมล่างซ้ายจะบอก mode ของการทำงาน

ข้อความที่เห็นจะเป็นส่วนหนึ่งของ file ที่เราเรียกขึ้นมา เนื่องจาก file มักจะมีขนาดใหญ่กว่าหน้าจอของเราอยู่มากเสมอ จึงทำการแบ่งเอาบางส่วนที่เราต้องการจะดูมาแสดงบนหน้าจอเท่านั้น ถ้าเราต้องการให้แสดงส่วนอื่นก็สามารถทำได้โดยใช้คำสั่งหรือปุ่มบังคับต่าง ๆ ดังจะอธิบายต่อไป แต่ก่อนอื่นเรามาศึกษาถึงการป้อนข้อมูลเข้าไปยัง file ที่เราต้องการก่อน

mode การทำงานของ Rand editor แบ่งออกได้เป็น 3 mode คือ

1. Editor mode เมื่อเรียกใช้ Rand โดยปกติจะเข้ามาอยู่ใน mode ที่นอกจากเรียกใช้ file ที่ผ่านมาก็จะอยู่ใน mode เก่าก่อนจะออกจากการทำงาน
ใน mode นี้ ด้านมุมล่างซ้ายจะไม่มีตัวบอกแสดงไว้ การป้อนข้อมูลก็จะเป็นแบบ overwrite โดยเมื่อกดปุ่มตัวอักษรใดก็จะเข้าไปแทนที่ตัวอักษรที่ตำแหน่ง Cursor
2. Insert mode จะเข้าสู่ mode นี้เมื่อกดปุ่ม <INS> และ เมื่อกดอีกทีก็จะออกจาก Insert mode สลับกันไป ด้านล่างซ้ายจะปรากฏคำว่า INSERT แสดง mode การทำงาน
ใน mode นี้จะเป็นการป้อนข้อมูลแบบแทรกเข้าไปยังตำแหน่งที่ Cursor ปรากฏ โดยต้นตัวอักษรไปทางขวา
3. Command mode เข้าสู่ command mode เมื่อกดปุ่ม <Alt> พร้อมกับ <A> และจะปรากฏข้อความแสดง mode การทำงานว่า
CMD:

ที่มุมล่างซ้าย

จากนั้นก็ทำการป้อน Command ต่าง ๆ ที่ต้องการเข้าไป กดปุ่ม <return> เป็นการสั่งให้ทำงาน และออกจาก command mode

จากการทำงานทั้ง 3 mode ที่ผ่านมาก็จะสามารถทำได้มีประสิทธิภาพถ้าเราสามารถเคลื่อนย้ายไปยังจุดที่ต้องการได้ ดังนั้นการควบคุมเคลื่อนย้าย cursor จึงเป็นเรื่องที่ควรที่จำเป็นจะต้องรู้ ปุ่มต่าง ๆ ที่เกี่ยวข้องกับ การควบคุม cursor ก็มีดังต่อไปนี้

- Arrow keys

เป็นกลุ่มคีย์ที่บังคับ Cursor ให้เคลื่อนไปซ้ายขวาบนและล่าง ก็จะประกอบด้วย คีย์ <<-- >, <-->>, <^>, < >

- Backspace key

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คีย์ปุ่ม <Backspace> เมื่อกดปุ่มนี้จะเลื่อน Cursor ไปยังทางซ้ายและลบตัวอักษรที่ตำแหน่งนั้น และดึงข้อความทางขวาเข้ามา

- Delete Character key

ได้แก่ปุ่ม จะทำการลบอักษรที่ตำแหน่ง Cursor หลังจากนั้นจะดึงข้อความทางขวาเข้ามาเช่นกัน

- Home key.

ได้แก่ปุ่ม <Home> จะย้าย Cursor ไปยังบนสุดด้านซ้ายของ window

- Carried return key

ได้แก่ปุ่ม <Enter> หรือ <CA> หรือ <<-- > จะเคลื่อนย้าย Cursor ไปยังตำแหน่งแรกของบรรทัดต่อไป

- Tabs key

ได้แก่ปุ่ม <Tab> จะเคลื่อนย้าย Cursor ไปตามตำแหน่งนี้ได้ set tab เอาไว้ โดยใช้คำสั่งใน Command mode ดังนี้

CMD : tab [n] <CR> set tab ตามตำแหน่ง n_1, n_2, \dots

CMD : tab s n <CR> set tab ให้หยุดเมื่อเคลื่อนไป n ตำแหน่ง

ถ้าจะยกเลิก tab ที่ตั้งไว้ก็ใช้คำสั่ง

CMD : -tab [n] <CR> ยกเลิกบางตำแหน่ง

CMD : -tab s <CR> ยกเลิกทุกตำแหน่ง

CMD : -tab s n <CR> ยกเลิกระยะที่ให้หยุด

- PgUp, PgDw key

ได้แก่ปุ่ม <PgUp>, <PgDw> โดยจะทำการเคลื่อนย้าย Cursor ไปส่วนต่างๆ บน file ที่ไม่ได้ปรากฏอยู่บน window โดยจะนำส่วนนั้นมาปรากฏบน window แทนเป็นจำนวนครึ่งละ 1 หน้าจอ

- goto command

ได้แก่คำสั่งใน Command mode ที่จะย้าย cursor ไปยังบรรทัดที่ต้องการโดยใช้คำสั่ง

CMD : goto [n] <CR> ไปยังบรรทัด n

CMD : goto b<CR> ไปยังต้น file

CMD : goto <CR> ไปยังท้าย file

CMD : goto e <CR> ไปยังท้าย file

การทำ Multiple windows

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Rand editor สามารถที่จะนำ file ขึ้นขึ้นมาอ่านบน window ที่ปรากฏอยู่ แล้วได้ คือสามารถที่จะดูหลาย file พร้อมกันได้ โดยเราเคลื่อนย้าย Cursor ไปยังตำแหน่งเริ่มต้นที่เราต้องการให้ window ของอีก file เริ่มต้น จากนั้นใช้คำสั่ง

CMD : window [filename] <CR>

ถ้าต้องการลบ window ใดที่ไม่ต้องการก็ใช้คำสั่ง

CMD : -window <CR> จะลบ window ล่าสุดที่สร้างขึ้น

นอกจากนั้น ยังสามารถที่จะทำการอ่าน file อื่นมาใช้นร่วมกับ window ที่ปรากฏอยู่โดยใช้คำสั่ง

CMD : e filename <CR>

และใช้คำสั่ง

CMD : e <CR> หรือปุ่ม <Alt>

เมื่อเลือกใช้ file ที่ต้องการปรากฏบน window

การ Mark และการทำงานกับส่วนที่ Mark

เราสามารถทำการ Mark ส่วนของ file ไปได้โดยจะกำหนดเป็น block เอาไว้ ทำได้โดยเคลื่อนย้าย cursor ไปยังจุดต้น block ที่เราต้องการ จากนั้นก็กดปุ่ม <F9> เมื่อเริ่มทำการ Block เคลื่อนย้าย cursor ไปยังจุดท้าย block ที่เราต้องการ เป็นอันว่าเราได้ block ที่ต้องการแล้ว ถ้าเราต้องการเลิก block ก็กดปุ่ม <F9> ซ้ำ อีกทีหนึ่ง

- การ insert blank lines

ทำได้โดยการกดปุ่ม <F1> จะทำการเพิ่ม blank line ไปยังบรรทัดที่ cursor ปรากฏอยู่และเลื่อนข้อความจากบรรทัดเก่าลงไปยังข้างล่าง หรือสามารถใช้คำสั่ง

CMD : open [n] <CR>

แต่ถ้ามีการ Mark block ไว้ก็จะทำการเติมบรรทัดว่างไว้ใน block นั้น แล้วเลื่อนข้อความที่อยู่ใน block ไปทางขวาหรือข้างล่าง

- การลบ line

โดยเราจะกดปุ่ม <F2> เพื่อลบบรรทัดที่ cursor ปรากฏอยู่ แล้วเลื่อนข้อความบรรทัดข้างล่างขึ้นมาแทน หรือใช้คำสั่ง

CMD : close [block] <CR>

ถ้ามีการ Mark block ก็จะมีการลบใน block นั้นแล้วเลื่อนส่วนอื่นเข้ามาแทน ถ้าต้องการเรียกส่วนที่เพิ่งลบไปกลับมาก็ใช้คำสั่ง

CMD : -Close <CR>

- การลบ text file

เมื่อเราไม่ต้องการใช้ file นั้นต่อไปก็จะทำการลบ file นั้นทิ้งได้โดยใช้คำสั่ง

CMD : delete <CR>

ถ้าต้องการเรียกกลับมาที่ออกจาก editor โดยใช้คำสั่ง

CMD : exit abort

- การลบข้อความ

จะทำการแทนค่าข้อความช่วงที่ต้องการลบด้วย space characters โดยใช้คำสั่ง

CMD : erase [block] <CR>

และมีการ Mark block นี้จะทำการลบภายใน block นั้น

ถ้าจะเรียกกลับคืนมาก็ใช้คำสั่ง

CMD : -erase <CR>

- การ copy

จะทำงาน 2 ขั้นตอนนี้ โดยจะเก็บส่วนของ file ที่ต้องการ copy ลงบน buffer โดยใช้คำสั่ง

CMD : pick [block] <CR>

รวมทั้ง ถ้ามีการ Copy Mark block ด้วย

หลังจากนั้นเลื่อน Cursor ไปยังตำแหน่งที่ต้องการจะนำส่วนของ file ไปไว้ แล้วใช้คำสั่ง

CMD : -pick <CR>

คำสั่งที่สำคัญ

ต่อไปจะแนะนำคำสั่งบางคำสั่งที่น่าสนใจและใช้บ่อย ๆ ในการสร้าง file ส่วนคำสั่งอื่น ๆ และการใช้งานคำสั่งต่าง ๆ อย่างละเอียดสามารถศึกษาได้จากคู่มือ Rand Editor ของระบบที่ใช้

- การแยกและต่อส่วนของข้อความ

ถ้าต้องการแยกก็เคลื่อนย้าย Cursor ไปยังจุดที่ต้องการแยกแล้วใช้คำสั่ง

CMD : split <CR>

ข้อความจากตำแหน่ง Cursor จะถูกแยกลงมาไว้ยังบรรทัดข้างล่าง โดยเลื่อนข้อความข้างล่างลงไป

ถ้าต้องการต่อข้อความหรือบรรทัดที่ใช้คำสั่ง

CMD : join <CR>

ก็จะทำการเลื่อนบรรทัดนี้ขึ้นไปต่อกับบรรทัดบน แล้วเลื่อนข้อความข้างล่างขึ้นมา

- การค้นหาและแทนค่า

เมื่อต้องการค้นหาและแทนค่านั้นก็สามารถใช้คำสั่งต่อไปนี้ได้

CMD : replace /text₁/text₂/ <CR> เริ่มที่ current cursor

CMD : -replace /text₁/text₂/ <CR> เริ่มที่ต้น file

โดยจะมี option ให้เลือกคือ

show จะแสดงการค้นหาและแทนค่าให้ดู

interactive สามารถเลือกได้ว่าจะแทนค่าหรือไม่

รูปแบบของคำสั่งจะได้ดังนี้

CMD : [-]replace[block] show /text₁/text₂/ <CR>
interactive

- การใช้คำสั่ง shell

เราสามารถที่จะใช้งานคำสั่งใน shell ได้ใน Rand editor โดยใช้คำสั่ง

CMD : run [block] shell-command <CR>

- การจัดการ file

เราสามารถที่จะเปลี่ยนชื่อ file ได้โดยใช้คำสั่งดังนี้

CMD : name filename <CR>

และเรายังทำการ save file ที่เราแก้ไขอยู่ได้โดยกดปุ่ม <END> หรือใช้

คำสั่ง

CMD : save filename <CR>

Rand editor จะทำการเก็บ file และสร้าง backup file โดยมี . นำหน้าชื่อ file

Screenlib เป็น tool ที่ใช้อำนวยความสะดวกในการจัดการเกี่ยวกับหน้าจอของ terminal ในระบบ UNIX ใช้งานรวมกับการโปรแกรมด้วยภาษา C เพื่อให้ใช้งานในระบบ UNIX ไร้ที่ติยิ่งขึ้น รูปแบบของการเขียนโปรแกรมที่จะใช้ Screenlib ช่วยในการจัดหน้าจอจะประกอบด้วยส่วนประกอบดังนี้

```
include <curses.h>
```

```
...
```

```
initscr();
```

```
cbreak();
```

```
noecho();
```

```
...
```

```
/* function ในภาษา C
```

```
และ screenlib */
```

```
...
```

```
endwin();
```

```
...
```

เมื่อจะใช้ screenlib โปรแกรมจะเรียกใช้ฟังก์ชันจาก `scrlib.o` library และฟังก์ชันส่วนมากมักจะต้องการใช้ `<curses.h>` ดังนั้น เริ่มแรกที่หัวโปรแกรมจึงต้องทำการ `include <curses.h>` เพื่อใช้ฟังก์ชันใน `scrlib.a` ใช้อ้างอิงได้ จากนั้นเมื่อจะเริ่มใช้ ส่วนที่เป็นฟังก์ชันใน `scrlib.a` ก็จะต้องเริ่มด้วยคำสั่ง `initscr();` เป็นการเริ่มต้นใช้งาน จากนั้นในส่วนต่อมาก็จะสามารถเรียกใช้ฟังก์ชันใน `scrlib.a` มาทำงาน รวมทั้งสามารถใช้ฟังก์ชันของภาษา C ได้ด้วย เมื่อจะเลิกใช้ฟังก์ชันใน `scrlib.a` แล้ว ก็จบด้วย `endwin();` เป็นการออกจากการใช้ `scrlib.a`

ในส่วนของการ compile โปรแกรมที่เรียกใช้ฟังก์ชันของ `scrlib.a` ก็จะใช้ `CC` ในการ compile เช่นกัน โดยมี option เพิ่มขึ้นดังนี้

- DSCRLIB

สำหรับในเรื่องของฟังก์ชันที่มีใช้งานใน screenlib นั้นเราจะอธิบายเฉพาะที่ใช้กันบ่อย ๆ และมีใช้ในการทำงานของโปรแกรมที่เขียนขึ้น โดยจะมีรายละเอียดดังนี้

1. การเริ่มต้นและการออกจาก screenlib

ในการใช้งาน screenlib output ของ screenlib นั้นแบ่งออกเป็น 2 ประเภทก็คือ

1. logical image ได้แก่ส่วนของ output ที่ฟังก์ชันในการทำงานได้ปฏิบัติด้วย
2. physical image ได้แก่ส่วน screen ที่เราเห็นจริง ๆ

ดังนั้นการกระทำใด ๆ ที่เกิดจากฟังก์ชันใน screenlib จะไปเกิดที่ logical image เพราะฉะนั้นเพื่อให้ผลมาปรากฏที่จอภาพให้เราสามารถที่จะเห็นได้ จึงต้องมีฟังก์ชันที่จะนำค่าต่าง ๆ ที่ปรากฏที่ logical image มาแสดงให้เห็นบน physical image ฟังก์ชันนั้นก็คือ refresh() ซึ่งมักจะใช้เมื่อต้องการให้ฟังก์ชันใดทำงานแล้วปรากฏที่ physical screen

จากรูปแบบของการเขียนโปรแกรมก็จะเห็นแล้วว่ามี การเริ่มต้นและลงท้ายที่เป็นแบบเฉพาะ โดยจะต้องเริ่มต้นด้วยฟังก์ชัน initscr() ซึ่งจะทำให้การ set ค่าเริ่มต้นของ screen ทั้ง logical และ physical image รวมทั้ง table ที่จำเป็นในการควบคุม screen ก่อนการใช้งานทุกครั้ง จึงต้องเรียก initscr() ก่อนจึงจะสามารถใช้งานฟังก์ชันอื่น ๆ ได้ ฟังก์ชัน initscr() จะให้ค่าเป็น 0 กลับมากำทำงานสำเร็จ ส่วนค่า -1 จะเกิดขึ้นเมื่อเกิด error

เมื่อทำงานที่เกี่ยวกับ screenlib เรียบร้อยแล้ว ก็จะใช้ฟังก์ชัน endwin() เป็นฟังก์ชันสุดท้ายก่อนออกจากโปรแกรม endwin() จะคืนสถานะแต่เดิมของ screen ปิด attributes ต่าง ๆ แล้วย้าย cursor ไปไว้มุมล่างซ้าย

2. การ Input/Output

ฟังก์ชันการ input และ output ใน screenlib กับ terminal ซึ่งปกติมักเป็นจอภาพกับคีย์บอร์ด สามารถกระทำได้ทั้งกับ Character และ String ซึ่งขึ้นกับ formatted ของ ฟังก์ชัน

ฟังก์ชันที่เราใช้นั้นจะกระทำกับ logical screen imageถ้าต้องการให้แสดงออกยังจอภาพก็ให้ใช้ฟังก์ชัน refresh() เพื่อให้ปรากฏบน physical screen ฟังก์ชันที่ใช้กันบ่อย ๆ ก็ได้แก่

addch(ch) เป็นการแสดงผล character ch
char ch

addstr(str) เป็นการแสดงผล string str
char *str;
printw (fmt, args) คล้ายกับ print f ในภาษา C โดยจะแสดงผล args ตาม format fmt ที่ให้ไว้
scanw (fmt, args) คล้ายกับ scan f ในภาษา C โดยจะอ่านผลจาก terminal ไปเก็บยัง args ตาม format fmt
getch () จะอ่านข้อมูลจาก terminal ครั้ง 1 ตัวอักษร แล้วเปลี่ยนเป็นค่า integer โดยจะมีค่าดังนี้

< octal 0400 : เป็น Character ปกติ

> octal 0400 : เป็น function key

และ cbreak mode อาจจะมี on ไว้

getstr (str) เป็นการเรียก getch() ทำงานจนกระทั่งอ่านถึงค่า newline เป็นการได้ string

3. การเคลื่อนย้าย Cursor

ฟังก์ชันที่ใช้ได้แก่

move (y, x)

int y, x;

โดย cursor จะถูกเคลื่อนย้ายไปที่บรรทัด y, ช่องที่ x ตำแหน่ง home หรือ top left corner คือ (0, 0)

ฟังก์ชัน move สามารถรวมกับฟังก์ชัน input/output ค่าที่ได้กล่าวมาแล้วได้ ก็จะเป็นการทำ input หรือ output ยังตำแหน่งที่กำหนดไว้ให้ ฟังก์ชันดังกล่าวได้แก่

mvaddch (y, x, ch)

mvaddstr (y, x, str)

mvprintw (y, x, fmt, args)

mvgetch (y, x)

mvgetstr (y, x, str)

mvscanw (y, x, fmt, args)

นอกจากนี้ยังมีฟังก์ชันการเคลื่อนย้ายอีก คือ

Cleft() เลื่อน cursor ไปทางซ้าย 1 ช่อง

Cright() เลื่อน cursor ไปทางขวา 1 ช่อง

4. การ set terminal

ฟังก์ชันที่ใช้บ่อย ๆ ได้แก่

`echo()`

`noecho()`

เป็นการบังคับให้ `input dcharacter` ที่ป้อนให้ปรากฏอยู่บน `terminal screen` หรือไม่ `echo()` จะปรากฏคำสั่ง `noecho()` จะไม่ปรากฏ

`Cbreak()`

`nocbreak()`

ถ้าใช้ `Cbreak` เมื่อ `input character` ที่เข้ามาจะผ่านไปยัง `program` ได้เลย แต่ถ้าใช้ `nocbreak` จะถูกเก็บไว้ใน `TTY driver` จนกระทั่งได้รับ `newline character`

5. ฟังก์ชันเกี่ยวกับหน้าจอ

ไว้แก่การ `save` หน้าจอและเรียกกลับคืน ได้แก่

`savscr()` จะทำการ `save` หน้าจอปัจจุบันไว้ โดยให้ค่าเป็น 0 ถ้าทำสำเร็จและค่าเป็น -1 ถ้าไม่สามารถทำได้

`rescr()` จะทำ `screen` ที่เก็บคำสั่งขึ้นมาแสดง โดยต้องสั่ง `refresh`

() เสียก่อน

การ `save` จะสามารถ `save` ได้ถึง 5 หน้าจอ

ฟังก์ชันการลบหน้าจอ ได้แก่

`clear()` จะเป็นการลบหน้าจอที่ปรากฏอยู่ออกไป

`erase_a(l1,c1,l2,c2,attr,delc)` เป็นการลบ `block` จากมุม (l1, c1) ไปยังมุม (l2,c2) โดยการเติม `character` ที่เราต้องการ ซึ่งมี `attribute` ตาม `attr.`

6. ฟังก์ชันเกี่ยวกับ Attribute

เป็นการกำหนด `attribute` ให้แก่ `character` โดยใช้

`blink()` .เป็นการซ่อนตัวอักษร

`reverse()` เป็นตัว `reverse`

under() เป็นตัวขีดเส้นใต้

or() เป็นการเลือกการกำหนดกลับไปเป็นเหมือนเดิม

นอกจากนี้ยังสามารถใช้ฟังก์ชันบางชนิดอีกด้วยเช่น

beep() จะส่งเสียง beep()

ฟังก์ชันทั้งหมดที่กล่าวมาเป็นฟังก์ชันการใช้งานที่ใช้อยู่ ๑ และมีใช้ในโปรแกรมของงานนี้เป็นส่วนมาก ซึ่ง screenlib ยังมีฟังก์ชันต่าง ๆ ที่สามารถใช้งานได้กว้างขวาง ถ้าสนใจที่จะใช้ก็สามารถหารายละเอียดได้จากคู่มือการใช้งาน screenlib ของระบบต่อไป



การออกแบบระบบโดยวิธี ISAC

การออกแบบระบบงานนั้นมีเทคนิคต่าง ๆ มากมายหลายรูปแบบ เทคนิคต่าง ๆ เหล่านี้เกิดขึ้นจากการสังสมประสบการณ์ของนักวิเคราะห์ระบบมาเป็นเวลานาน ความผิดพลาดและความสำเร็จของการออกแบบระบบ จะช่วยให้นักวิเคราะห์ระบบเกิดแนวความคิดว่า การออกแบบระบบที่ดีควรทำอย่างไร ควรคำนึงถึงสิ่งใด และควรกำหนดขั้นตอนอย่างไรบ้าง

ISAC หรือ Information Systems Work and Analysis of Changes เป็นวิธีการออกแบบระบบวิธีหนึ่ง ที่คิดค้นโดยกลุ่มนักวิเคราะห์ระบบชาวสวีเดน และใช้กันอย่างแพร่หลายตามบริษัทใหญ่ ๆ ทางกลุ่มประเทศสแกนดิเนเวีย [1]

การออกแบบระบบโดยวิธี ISAC เป็นการวิเคราะห์และพัฒนาระบบงานใด ๆ ที่เน้นทางด้านการวิเคราะห์และการออกแบบระบบ โดยมีจุดมุ่งหมายที่จะนำคอมพิวเตอร์เข้าไปช่วยปรับปรุงการทำงาน หรือกิจกรรมต่าง ๆ ขององค์การ ให้เกิดความสะดวกรวดเร็ว และมีประสิทธิภาพเพิ่มมากขึ้น แต่ก่อนที่จะทำการวิเคราะห์และพัฒนาระบบได้นั้น จำเป็นจะต้องศึกษากิจกรรมขององค์การให้เกิดความเข้าใจเสียก่อน

1. กิจกรรมขององค์การ

งานส่วนแรกของ ISAC คือพยายามทำความเข้าใจกับกิจกรรมต่าง ๆ ขององค์การก่อนที่จะก้าวไปถึงขั้นการวิเคราะห์และพัฒนาระบบ นั่นคือก่อนที่นักวิเคราะห์จะทำการวิเคราะห์และพัฒนาระบบได้นั้น จำเป็นต้องทราบถึงระบบงานขององค์การทั้งหมดที่กำลังดำเนินอยู่ รวมไปถึงกิจกรรมและขั้นตอนการทำงานของกิจกรรมนั้น ๆ เสียก่อน เพื่อนำมาใช้ประกอบการนิยามหาทางแก้ไขปัญหาที่เกิดขึ้นกับระบบปัจจุบัน รวมไปถึงการตอบสนองความต้องการของผู้ใช้ระบบต่อไป

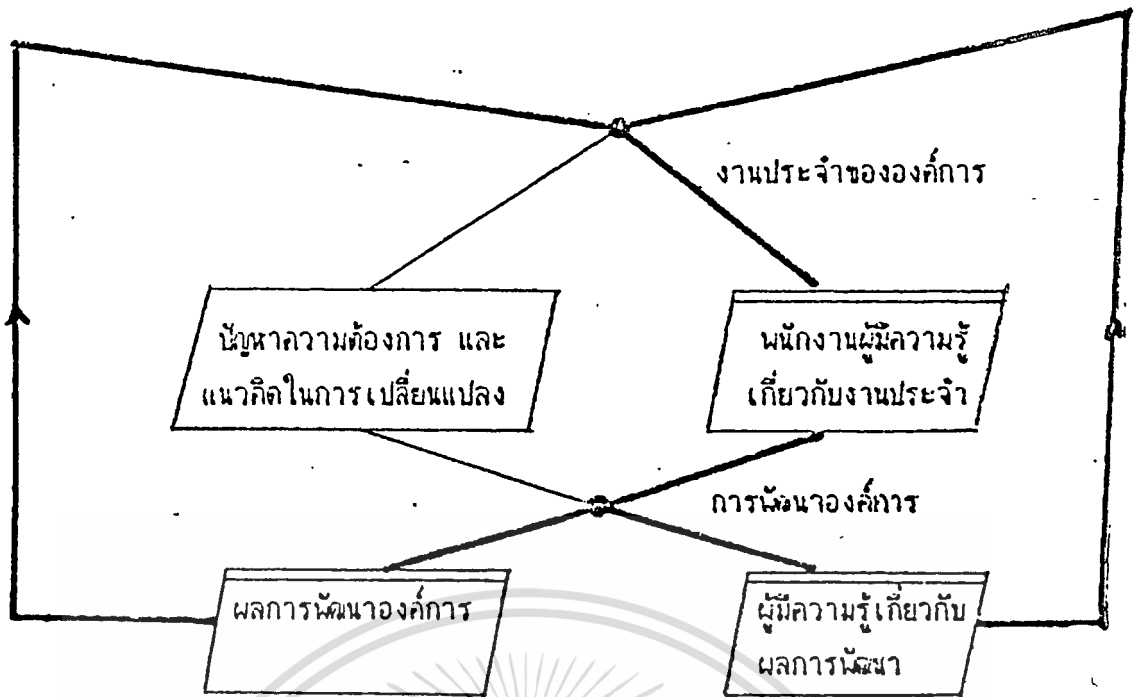
กิจกรรมต่าง ๆ ในองค์การนั้น อาจจำแนกได้เป็นสองส่วน ส่วนแรกคือ งานประจำขององค์การ เป็นงานธรรมดาที่ทำกันอยู่ประจำวัน ส่วนที่สองคืองานพัฒนาองค์การ เป็นงานที่เกี่ยวข้องกับการสร้างงาน หรือ กิจกรรมในส่วนแรก ซึ่งงานในส่วนนี้เป็นหัวใจสำคัญที่จะนำไปสู่การวิเคราะห์และพัฒนาระบบต่อไป

เราสามารถแสดงกิจกรรมทั้งสองประเภทได้ดังรูปที่ 1 ซึ่งเป็นแผนภูมิกิจกรรม หรือ Activity Graph (A-Graph)

เมื่อพิจารณาการวิเคราะห์และพัฒนาระบบตามเค้าโครงของรูปที่ 1 แล้ว จะเห็นปรัชญาการออกแบบระบบ ISAC ได้ชัดเจนยิ่งขึ้น กล่าวคือ การทำงานในองค์การนั้นย่อมมีปัญหาขัดข้องหลายประการ ดังนั้นจึงต้องพัฒนาระบบใหม่เพื่อจัดหรือแก้ปัญหาข้างต้นให้หมดไป แต่ระบบใหม่ที่สร้างจะได้ผลดีต่อเมื่อมีการวิเคราะห์ปัญหาของระบบเดิมอย่างถี่ถ้วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1 แผนภูมิกิจกรรมแสดงกิจกรรมในองค์การ

1.1 แผนภูมิกิจกรรม

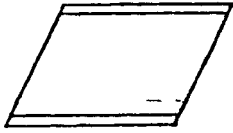
หน้าที่ของแผนภูมิกิจกรรม คือ ใช้ในการอธิบายกิจกรรมในหน่วยงานและระบบสารสนเทศที่เกี่ยวข้อง สัญลักษณ์ที่ใช้ในแผนภูมิกิจกรรมแทนสิ่งต่าง ๆ ต่อไปนี้

1. เซต (Set) ได้แก่ เซตของสิ่งต่าง ๆ ทั้งบุคคล(Person), วัสดุ(Material) และข่าวสาร(Message) เซตที่เคลื่อนเข้าสู่กิจกรรมเราเรียกว่า อินพุต ส่วนเซตที่เคลื่อนออกจากกิจกรรมเราเรียกว่า เอาต์พุต ของกิจกรรม
2. กิจกรรม (Activity) กิ่งที่กระทำโดยมนุษย์, เครื่องมืออุปกรณ์ หรือประกอบกัน
3. การเคลื่อนที่ (Flow) แสดงการไหลของเซตระหว่างกิจกรรมต่าง ๆ

คำอธิบายสัญลักษณ์ที่ใช้ในแผนภูมิกิจกรรมมีอธิบายในรูปที่ 2

1.1.1 ตัวอย่างของแผนภูมิกิจกรรม

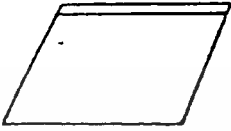
แผนภูมิกิจกรรมตัวอย่างเขียนขึ้นจากระบบงานของสหกรณ์ผู้ผลิตนมแห่งหนึ่ง ซึ่งรวมตัวกันขึ้นจากผู้ผลิตหลายราย สหกรณ์นี้มีศูนย์อยู่ที่ออฟฟิศกลางอันเป็นสถานที่ประชุมวางแผนการดำเนินกิจการ ทางผู้ผลิตแต่ละรายจะรับผิดชอบการผลิตและการจำหน่ายเองการขายปลีกนั้นจะผ่านพ่อค้าคนกลาง ฝ่ายสหกรณ์จะมีการติดต่อสื่อสารกับผู้บริหารในด้านข่าวสาร เช่น การโฆษณา เป็นต้น ระบบที่เราสนใจ (ระบบที่เกิดปัญหาและต้องแก้ไข) ได้แก่ระบบการรับคำสั่งซื้อและการส่งสินค้าไปยังลูกค้า จากสิ่งต่าง ๆ ดังกล่าวเราเขียนแผนภูมิกิจกรรมได้ดังรูปที่ 3 ซึ่งมีรายละเอียดบางอย่างที่ยังไม่ได้กล่าวถึง ดังนี้คือ



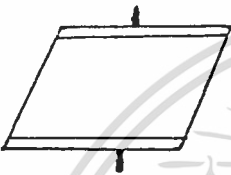
เชื่ตจริง
เชื่ตของบุคคล และ/หรือวัสดุ



เชื่ตข่าวสาร
เชื่ตของข่าวสาร เช่น เอกสาร
หรือการรับข่าวทางโทรทัศน์



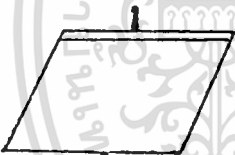
เชื่ตประกอบ
เชื่ตของบุคคล วัสดุและข่าวสาร



การเคลื่อนที่จริง
การเคลื่อนที่ของบุคคล/วัสดุ



การเคลื่อนที่ข่าวสาร



การเคลื่อนที่ประกอบ
การเคลื่อนที่ของบุคคล/วัสดุและข่าวสาร

กิจกรรม

หมายเหตุ การเคลื่อนที่ทั้งหมดถือว่าเป็นการเคลื่อนที่จากบนลงล่าง ดังนั้นจึงไม่ต้องเขียนหัวลูกกร จะเขียนเฉพาะเมื่อเป็นการเคลื่อนที่จากล่างขึ้นบนเท่านั้น
รูปที่ 2 สัญลักษณ์ที่ใช้ในแผนภูมิกิจกรรม

1. บนกระดาษที่เขียนแผนภูมิกิจกรรมจะมีหัวกระดาษระบุสิ่งที่ควรรู้ โดยแบ่งออกเป็น 3 คอลัมน์ อันได้แก่

คอลัมน์ที่ 1 ทางซ้ายมือสุด แยกเป็น 3 ส่วนย่อยอีก คือ

- ส่วนที่ 1 บอกชื่อของหน่วยงาน, องค์กร ในที่นี้ สมมติเป็นสมณ. (สหกรณ์ผู้ผลิตนม)
- ส่วนที่ 2 บอกชื่อบุคคลผู้วิเคราะห์ระบบ (หรือกลุ่ม)
- ส่วนที่ 3 บอกขอบเขตระบบที่แผนภูมิกิจกรรมแสดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอลัมน์ที่ 2 ตรงกลาง แยกเป็น 2 ส่วนย่อย

- ส่วนที่ 1 บอกให้ทราบว่าเอกสารนี้ คือ-แผนภูมิกิจกรรม
- ส่วนที่ 2 บอกวันที่ที่เขียนหรือปรับปรุง (ตามเวอร์ชัน)

คอลัมน์ที่ 3 ทางขวามือสุด แยกเป็น 2 ส่วนย่อย

- ส่วนที่ 1 บอกเลขอ้างอิงของแผนภูมิกิจกรรม โดยบ่งถึงกิจกรรมที่แผนภูมิกิจกรรมนั้นอธิบาย
- ส่วนที่ 2 บอกเวอร์ชันที่ปรับปรุง

2. กรอบสี่เหลี่ยมที่ล้อมรอบให้กิจกรรมทั้งหลายอยู่ใน ก็คือขอบเขตของระบบงานที่เรากำลังศึกษา กิจกรรมทั้งหลายที่เราสนใจเป็นกิจกรรมภายในระบบงานจึงอยู่ภายในกรอบสี่เหลี่ยม แต่สำหรับเซตจะมีอยู่ 3 จำพวก พวกแรกเป็นเซตที่เกิดจากกิจกรรมภายนอกระบบ แต่เข้ามาเป็นอินพุตของกิจกรรมภายในระบบของเรา เซตพวกนี้จะอยู่ด้านบนนอกกรอบสี่เหลี่ยม ตามตัวอย่างได้แก่ เซต 1A และเซต 2A ต่อมาก็คือพวกที่เป็นเอาต์พุตหรือเกิดจากกิจกรรมภายในระบบ และยังคงใช้งานภายในระบบ (คือเป็นอินพุตของกิจกรรมอื่นอีกทีหนึ่ง) พวกนี้จะอยู่ภายในกรอบสี่เหลี่ยม ดังเช่น เซต 3A เซต 5B เป็นต้น สำหรับพวกสุดท้ายเป็นเซตที่เกิดจากกิจกรรมในระบบเช่นเดียวกับพวกที่สอง แต่พ้นจากระบบไปเป็นอินพุตของกิจกรรมในระบบอื่น ภายนอก เซตพวกนี้จะอยู่ด้านล่างนอกกรอบสี่เหลี่ยม ตามตัวอย่างได้แก่ เซต 3C และเซต 6A

3. แต่ละกิจกรรมภายในระบบจะมีหมายเลขเพื่อใช้อ้างอิงที่แตกต่างกันและมีชื่อบ่งบอกไว้ด้วยเช่น กิจกรรมหมายเลข 3 คือกิจกรรมในออฟฟิศกลาง กิจกรรมหมายเลข 6 คือกิจกรรมของผู้บริโภค เป็นต้น สำหรับเซตต่างๆก็มีตัวเลขและตัวอักษรใช้อ้างอิงเช่นเดียวกัน การกำหนดเลขอ้างอิงของกิจกรรมให้กำหนดเลขเรียงลำดับกันจากน้อยไปมากโดยไล่เรียงกิจกรรมที่มีตำแหน่งบนแผนภูมิกิจกรรมจากบนลงล่าง และจากซ้ายไปขวา ส่วนเซตมีหลักเกณฑ์คือ ถ้าเซตนั้นเป็นเอาต์พุตของกิจกรรมใด ก็ใช้เลขอ้างอิงของกิจกรรมนั้นประกอบด้วยตัวอักษรภาษาอังกฤษ A, B, ... เป็นตัวอ้างอิงเซต ตัวอย่างเช่น กิจกรรมหมายเลข 3 มีเอาต์พุต 3 เซต ก็จะได้ 3A, 3B และ 3C เป็นตัวอ้างอิงเป็นต้น โดยที่เราไม่ต้องสนใจว่า เซตนั้นอยู่ภายในกรอบสี่เหลี่ยมหรืออยู่นอกกรอบสี่เหลี่ยมด้านล่าง ส่วนเซตที่อยู่ภายนอกกรอบสี่เหลี่ยมด้านบนนั้น ให้ถือเสมือนว่ามันเป็นเอาต์พุตของกิจกรรมอะไรก็ได้ภายนอกระบบ และกำหนดตัวอ้างอิงขึ้นเองโดยใช้หลักเดียวกับที่กล่าวมาแล้ว แต่ห้ามใช้ตัวอ้างอิงที่ซ้ำซ้อน จากตัวอย่างในรูปเช่น เซต 2A ก็ถือเสมือนว่ามันเป็นเอาต์พุตของกิจกรรมหมายเลข 2 ภายนอกระบบ เป็นต้น

4. ที่มุมบนซ้ายของกรอบสี่เหลี่ยมจะนำเลขอ้างอิงของแผนภูมิกิจกรรม (ซึ่งเขียนอยู่ในกรอบสี่เหลี่ยมเล็ก ขุนหวักระดาศ) มาเขียนกำกับไว้ ตามตัวอย่างนั้น เลขอ้างอิงของแผนภูมิกิจกรรม คือ C0 ก็เขียนตัว C กำกับ(ถ้าเป็นเลข 0 มักจะละไว้ ไม่ลงเป็น C0)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สผน.

แผนภูมิกิจกรรม

ผู้วิเคราะห์ :

กลุ่มพัฒนาระบบสารสนเทศ

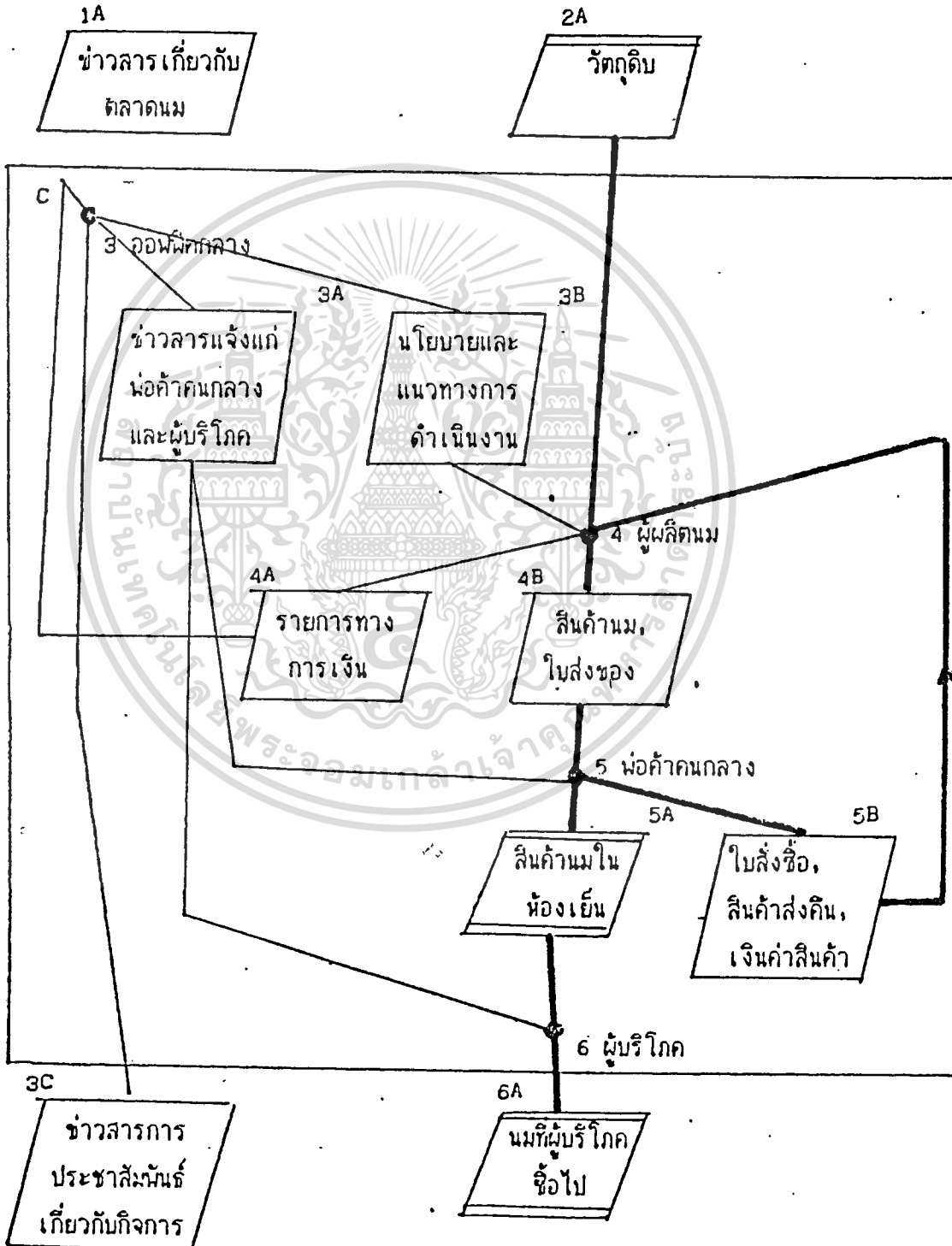
วันที่ : 9 เม.ย. 2533

เลขอ้างอิง C0

ขอบเขต :

ระบบรับคำสั่งซื้อและส่งสินค้า

เวอร์ชัน 4.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3 แผนภูมิกิจกรรมขององค์กรผู้ผลิตนม นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สผน.

รายการกำกับ

ผู้วิเคราะห์ :

แผนภูมิกิจกรรม

กลุ่มพัฒนาระบบสารสนเทศ

เลขอ้างอิง C0

ขอบเขต :

วันที่ : 9 เม.ย. 2533

ระบบรับคำสั่งซื้อและส่งสินค้า

เวอร์ชัน 4

1A ข่าวสารเกี่ยวกับตลาดนม

1A1 ข่าวสารทางสื่อมวลชน

1A2 ข่าวสารจากแหล่งอื่น

2A วัตถุประสงค์

2A1 ฐานวัตถุดิบ

2A2 ส่วนผสม เช่น น้ำตาล เป็นต้น

2A3 บรรจุภัณฑ์

3 การวางแผนและแนวทางการดำเนินงาน ณ ออฟฟิศกลาง

3A ข่าวสารที่ส่งให้กับพ่อค้าคนกลางและผู้บริโภคร

3A1 โฆษณาสำหรับพ่อค้าคนกลาง (ไปยังกิจกรรมหมายเลข 5)

3A2 ข่าวเรื่องการจัดคอมพิวเตอร์ส่งเสริมการขาย (ไปยังกิจกรรมหมายเลข 5)

3A3 โฆษณาสำหรับผู้บริโภค เช่น โคเร็กซ์เมล์ (ไปยังกิจกรรมหมายเลข 6)

3B แผนงานและแนวทางการดำเนินงาน

3B1 แผนงาน

3B2 แนวทางการดำเนินงาน

4 การผลิตและจำหน่ายของผู้ผลิตนม

4A รายงานทางการเงินของผู้ผลิตนม

4B สินค้าที่ผลิตแล้วและใบส่งของที่จะส่งไปให้พ่อค้าคนกลาง

4B1 นมสำเร็จรูป

4B2 ใบส่งของ

เอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 การรับและเก็บสินค้า ณ ที่ประกอบการของพ่อค้าคนกลาง

- 5A สินค้า (นม) ที่เก็บเอาไว้ในห้องเย็น รอให้ผู้บริโภคซื้อไป
- 5B ค่าสั่งซื้อ, เงินค่าสินค้า และสินค้าที่ถูกส่งกลับคืน
- 5B1 ค่าสั่งซื้อ อาจเป็นใบสั่งซื้อหรือสั่งทางโทรศัพท์
- 5B2 เงินชำระค่าสินค้า หลังจากพ่อค้าคนกลางได้รับสินค้าแล้ว
- 5B3 สินค้าที่เงื่อนไขไม่ตรงตามที่ส่งถูกส่งคืน

6 การซื้อนมของผู้บริโภค

- 3C ข้าราชการและประชาสัมพันธ์กิจการ
- 3C1 ข้าราชการที่ให้กับสื่อมวลชน
- 3C2 ข้าราชการให้ผู้รับอื่นๆ
- 6A นมที่ซื้อไปบริโภค

รูปที่ 4 ตัวอย่างรายงานกำกับ

ข้อกำหนดในการเขียนรายงานกำกับ มีเป็นข้อ ๆ ดังนี้

1. รายงานกำกับมีหัวกระดาษเช่นเดียวกับแผนภูมิกิจกรรม แต่เพิ่มคำว่ารายการกำกับ ตรงกลางหัวกระดาษ

2. ตัวเนื้อหาของรายการกำกับแบ่งเป็น 3 ส่วน แต่ละส่วนแยกให้เห็นชัดจากกันด้วยเส้นตรงขนาน โดยที่

- ส่วนที่ 1 อธิบายรายละเอียดของสิ่งที่อยู่นอกกรอบสี่เหลี่ยมด้านบนของแผนภูมิกิจกรรม ในที่นี้ได้แก่ เซ็ต 1A และเซ็ต 2A
- ส่วนที่ 2 อธิบายรายละเอียดของสิ่งที่อยู่ภายในกรอบสี่เหลี่ยมของแผนภูมิกิจกรรม ในที่นี้ได้แก่ กิจกรรมหมายเลข 3, 4, 5 และ 6 ตลอดจนเซ็ต 3A, 3B, 4A, 4B, 5A และ 5B
- ส่วนที่ 3 อธิบายรายละเอียดของสิ่งที่อยู่ภายนอกกรอบสี่เหลี่ยมด้านล่างของแผนภูมิกิจกรรม ในที่นี้ได้แก่ เซ็ต 3C และเซ็ต 6A

3. แต่ละเซ็ตภายในแผนภูมิกิจกรรมสามารถแบ่งเป็นเซ็ตย่อยเพื่อเจาะลึกถึงรายละเอียดของเซ็ตนั้นได้ (โดยที่เซ็ตย่อยเหล่านั้นไม่ได้แสดงไว้ในแผนภูมิกิจกรรม) ตัวอย่างตามรูปที่ 3 เซ็ต 1A แบ่งได้เป็นเซ็ต 1A1 และเซ็ต 1A2 ส่วนเซ็ต 2A ก็แบ่งเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำอธิบายระบบงานโดยอาศัยแผนภูมิกิจกรรมประกอบ

จากแผนภูมิกิจกรรมในรูปที่ 3 ออഫิศกลาง (กิจกรรม 3) จะรับรู้เอาข่าวสารเกี่ยวกับตลาดนมจากภายนอกระบบ (เซต 1A) ข่าวสารนี้อาจได้จากสื่อมวลชน, ผู้ผลิตนมรายอื่นหรือจากแหล่งข่าวอื่น และยังได้รับรายงานทางด้านการเงินจากสมาชิกของสหกรณ์ (4A) ทางออฟิศกลางจะใช้อินพุตทั้งสองเป็นเครื่องช่วยกำหนดแผนและแนวทางดำเนินงานให้กับสมาชิก (3B) และมีการแจ้งข่าวสารความเคลื่อนไหวของกิจกรรมต่างๆ ไปยังลูกค้า คือผู้บริหารและพ่อค้าคนกลาง (3A) รวมทั้งข่าวสารประชาสัมพันธ์ไปยังภายนอกระบบ (3C) เช่น ให้ข่าวแก่หนังสือพิมพ์ธุรกิจ เป็นต้น ทางด้านผู้ผลิตนม (4) จะได้รับวัตถุดิบในการผลิต เช่น น้านมดิบ, ส่วนผสม (เช่น น้าตาล) และบรรจุภัณฑ์ เป็นต้น (2A) ตลอดจนรับแผนงานและแนวทางดำเนินงานจากออฟิศกลาง (3B) นอกจากนี้ยังมีคำสั่งซื้อ, เงินค่าสินค้าและสินค้าส่งกลับคืนจากพ่อค้าด้วย (5B) และแน่นอนสิ่งที่ได้ออกมาจากกิจกรรมนี้ก็คือ สินค้านมที่ผลิตแล้ว ส่งไปยังพ่อค้าพร้อมกับใบส่งของ (4B) และส่งรายงานด้านการเงินกลับไปยังออฟิศกลาง (4A) ไล่ลงมายังพ่อค้าคนกลาง (5) ได้รับสินค้ากับใบส่งของจากผู้ผลิตนม (4B) และรับข่าวสารจากออฟิศกลาง เช่น การจัดแคมเปญส่งเสริมการขาย เป็นต้น (3A) แล้วทางพ่อค้าจะส่งนมไปยังผู้บริหาร (5A) และส่งคำสั่งซื้อ, เงินชำระค่าสินค้า และสินค้ากลับคืนไปยังผู้ผลิต (5B) สุดท้าย ได้แก่ ผู้บริโภค (6) ได้รับข่าวสาร เช่น โฆษณาสินค้าได้เรีกัดเมล็ดจากออฟิศกลาง (3A) โดยที่ จะมีนมที่ไปซื้อมาไว้บริโภค (6A)

1.1.2 รายงานกำกับ (TEXT PAGE)

ในการใช้งานจริง ๗ แล้วแผนภูมิกิจกรรมยังมีอุปสรรคและข้อจำกัดอยู่ เพราะหน้ากระดาษมีเนื้อที่จำกัด ผู้เขียนแผนภูมิกิจกรรมจะถูกบังคับทางอ้อม 2 ประการ ได้แก่ ประการแรก การเขียนข้อความใด ๆ ต้องให้สั้นและกะทัดรัดเข้าไว้ อย่างเช่น กิจกรรมหมายเลข 5 ต้องใช้คำว่า "ผู้ผลิตนม" แทนข้อความ "การผลิตและจำหน่ายสินค้าของผู้ผลิตนม" เป็นต้น ส่วนประการที่สองก็คือ หากสิ่งที่เราสนใจ (เซต) มีมาก จะต้องพยายามรวมสิ่งเหล่านั้นเข้าไว้เป็นเซตเดียวกัน เมื่อไม่มีจำนวนเซตน้อยที่สุด เช่น เซต 5B แทนที่จะแยกเป็นเซตของคำสั่งซื้อ เซตของเงินค่าสินค้า และเซตของสินค้าส่งคืน ก็ต้องนำมารวมเป็นเซตเดียว ข้อจำกัดดังกล่าวส่งผลให้แผนภูมิกิจกรรมอาจรวบรัดเกินไป จนผู้ทำความเข้าใจได้ยาก กลุ่มไอเอ็มซีจึงได้ตกลงกำหนดให้มีการทำเอกสารเสริมขึ้นมาชุดหนึ่ง เรียกว่า รายงานกำกับ ใช้สำหรับแจกแจงรายละเอียดของเซตและของกิจกรรมในแผนภูมิกิจกรรมว่า มีเซตและกิจกรรมอะไรบ้าง โดยมีข้อความอธิบายเพิ่มเติมจากข้อความที่เขียนไว้สั้น ๆ ในแผนภูมิกิจกรรม และถ้าเซตใดมีส่วนประกอบที่ต้องการแจกแจง ก็อาจเขียนแยกเป็นเซตย่อย (subset) ไว้ให้ ตัวอย่าง Text Page ได้แสดงไว้ดังรูปที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช็ท 2A1, 2A2 และ 2A3 แต่ไม่ได้แสดงไว้ มาแจกแจงไว้ที่รูปที่ 4 การแบ่งเช็ทย่อย จะแบ่งเท่าไรก็ขึ้นอยู่กับผู้เขียน แผนภูมิกิจกรรมเอง บางเช็ทอย่างเช่นเช็ท 4A และเช็ท 5A อาจไม่ต้องแบ่งก็ได้

4. ในการเขียนจะเรียงลำดับหมายเลขอ้างอิงของเช็ทและของกิจกรรมจากน้อยไป มากภายในส่วนเดียวกัน (หมายถึง ส่วนทั้งสามที่ได้อธิบายไปในข้อ 2) ตามรูปที่ 4 นั้น ส่วนที่ 1 เขียน 1A ก่อน 2A ส่วนที่ 3 ก็เขียน 3C ก่อน 6A สำหรับส่วนที่ 2 จะเขียน กิจกรรมที่มีเลขอ้างอิงน้อยก่อน ตามด้วยเช็ทที่เป็นเอาต์พุตของกิจกรรมนั้น แล้วจึงเขียน กิจกรรมที่มีหมายเลขอ้างอิงถัดไปเรื่อย ๆ (โปรดดูตัวอย่างประกอบ)

5. จะสังเกตเห็นว่าเวลาเขียนถึงเช็ทและกิจกรรมที่มีอ้างอิงไว้ในแผนภูมิกิจกรรม จะ มีการเว้นบรรทัดเสมอ เฉพาะเช็ทย่อยซึ่งเราไม่ได้อ้างอิงไว้ในแผนภูมิกิจกรรมจึงจะเขียนอยู่ ได้เช็ทใหญ่ของมันโดยไม่มีการเว้นบรรทัด จะเห็นว่ารายการกำกับมีประโยชน์ต่อแผนภูมิ กิจกรรมอย่างมากในแง่ของการเป็นตัวเสริมให้แผนภูมิกิจกรรมมีประสิทธิภาพในการสื่อ ความหมายให้ผู้ดูสามารถเข้าใจระบบได้กระจ่างขึ้น

1.1.3 แผนภูมิสรุปและแผนภูมิรายละเอียด (Overview Graph and Detail Graph)

จุดเด่นประการหนึ่งของแผนภูมิกิจกรรม คือสามารถจำลอง หรืออธิบายการทำงาน ของระบบได้ละเอียดมากเท่าที่ผู้วิเคราะห์ต้องการ แผนภูมิกิจกรรมที่ถูกเขียนขึ้นมาแรกสุด และมีความละเอียดน้อยที่สุดก็คือ แผนภูมิสรุป จากนั้นถ้าผู้วิเคราะห์ต้องการอธิบายส่วน ย่อยต่าง ๆ ให้ชัดเจนขึ้นอีก ก็สามารถจะกระทำได้โดยการเขียนแผนภูมิรายละเอียด ให้ละ ละเอียดลงไปเรื่อย ๆ เป็นลำดับไป มีข้อสังเกตอยู่ว่าแผนภูมิสรุปของแต่ละระบบจะมีเพียง ระดับเดียวและมีเพียงภาพเดียว แต่แผนภูมิรายละเอียดอาจจะมีได้หลายระดับ และแต่ละ ระดับก็อาจมีได้หลายภาพ ตัวอย่างของแผนภูมิรายละเอียดได้แสดงไว้ดังรูปที่ 5

จากแผนภูมิสรุปของระบบรับคำสั่งซื้อและส่งสินค้า ต้องการขยายรายละเอียดของกิจ กรรมหมายเลข 4 "ผู้ผลิตนม" จะเขียนแผนภูมิกิจกรรมและรายการกำกับได้ดังรูปที่ 6 ซึ่ง จะอธิบายดังนี้ กรอบสี่เหลี่ยมใหญ่ที่เป็นเช่นเดียวกับแผนภูมิสรุป ก็เป็นการแสดงขอบ เขตของระบบที่สนใจ ซึ่งตามรูปนี้ขอบเขตดังกล่าวคือ กิจกรรมหมายเลข 4 "ผู้ผลิตนม" ของแผนภูมิสรุปภายในกรอบแบ่งเป็นกิจกรรมของแผนกต่าง ๆ นั้นหมายความถึงการแบ่ง แผนกของผู้ผลิตนม ซึ่งที่จริงจะแบ่งกิจกรรมเป็นอย่างไรก็ได้ เช่นอาจรวมหลายแผนก เป็นกิจกรรมเดียวกัน หรือแผนกเดียวแต่แยกเป็นหลายกิจกรรม ทั้งนี้ขึ้นอยู่กับวิจาร์แผน ของผู้วิเคราะห์ นอกจากนี้ดังได้กล่าวไว้แล้วว่าแผนภูมิรายละเอียดมีได้หลายระดับ ดังนั้น ผู้วิเคราะห์ยังอาจขยายรายละเอียดเพิ่มได้อีก เช่น นำกิจกรรมของแผนกเก็บรักษาและส่ง สินค้าไปเขียนเป็นแผนภูมิกิจกรรมภาพใหม่ ซึ่งจะ เป็นแผนภูมิรายละเอียดระดับที่ 2 เป็น ต้น แต่ในที่นี้เราจะไม่เจาะรายละเอียดเหล่านั้นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปนี้ได้แบ่งการพัฒนาองค์การออกเป็นสองประเภท ประเภทหนึ่งคือการวิเคราะห์ และพัฒนาระบบคอมพิวเตอร์อันเป็นเรื่องที่จะทำการศึกษากันต่อไป ส่วนอีกประเภทหนึ่งคือ การพัฒนาอื่น ๆ อันได้แก่ การพัฒนาผลิตภัณฑ์หรือการพัฒนามนุษย์ แต่ก่อนที่จะทำการพัฒนาระบบคอมพิวเตอร์หรือพัฒนาด้านอื่น ๆ นั้น เราจะต้องวิเคราะห์ปัญหาและความต้องการเสียก่อน โดยทำการวิเคราะห์การเปลี่ยนแปลง (Change analysis) และดูว่าความต้องการนั้น ๆ เหมาะสมหรือไม่ ถ้าหากไม่เหมาะสมหรือไม่สามารถแก้ไขได้ ก็ต้องกลับไปวิเคราะห์ใหม่อีก ดังแสดงด้วยลูกศรสั้น ๆ ย้อนกลับขึ้นข้างบน

หลังจากทำการวิเคราะห์การเปลี่ยนแปลงเสร็จเรียบร้อยแล้ว งานที่ต้องต่อไปก็คือ การวิเคราะห์และพัฒนาระบบคอมพิวเตอร์ ที่แบ่งกิจกรรมออกเป็นสองขั้นตอนดังนี้คือ ขั้นที่หนึ่งเป็นการวิเคราะห์และออกแบบระบบสารสนเทศ (Analysis and Design of information systems) ขั้นต่อมาเป็นการจัดทำระบบและใช้งาน (Realization of information systems and implementation)

เมื่อพิจารณากิจกรรมตามเค้าโครงของรูปที่ 7 สามารถสรุปขั้นตอนในการออกแบบระบบ หลังจากทำการศึกษากิจกรรมขององค์การ เรียงตามลำดับได้ดังนี้

- วิเคราะห์ความเปลี่ยนแปลง (Change analysis)
- วิเคราะห์และออกแบบระบบ (Analysis and Design)
- จัดสร้างระบบและใช้งาน (Realization and Implementation)

2.1 การวิเคราะห์ความเปลี่ยนแปลง

ก่อนการวิเคราะห์ระบบนั้น เราต้องวิเคราะห์ความเปลี่ยนแปลงเพื่อหาว่า เราควรคิดเปลี่ยนแปลงอะไรบ้าง เพื่อแก้ปัญหาที่เกิดขึ้นกับระบบปัจจุบัน และการเปลี่ยนแปลงนั้นจะก่อให้เกิดผลสิ่งที่มีต้องการหรือไม่ การวิเคราะห์การเปลี่ยนแปลงนี้มีขั้นตอนต่าง ๆ ดังแสดงในรูปที่ 8

แต่ก่อนที่จะทราบถึงวิธีในการวิเคราะห์ความเปลี่ยนแปลง เราควรมาทำความรู้จักกับเทคนิคที่ใช้ในการบรรยาย (Description technique) เสียก่อน เทคนิคในการบรรยายประกอบด้วยส่วนสำคัญสามส่วน คือ

- แผนภูมิกิจกรรม (A-graphs)
- รายการกำกับ (Text pages)
- ตารางแสดงคุณสมบัติ (Property tables)

สองส่วนแรกได้มีการกล่าวไว้ก่อนแล้วในหัวข้อ กิจกรรมขององค์การ

ตารางแสดงคุณสมบัติ

จากแผนภูมิกิจกรรมและเทคนิคเหล่านี้ เราสามารถแสดงขั้นตอน, ลักษณะการดำเนินงาน แต่ไม่สามารถแสดงจำนวน, ปริมาณหรือขนาดของการเคลื่อนที่, จำนวนเอกสารที่เข้าและเอกสารนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1 การวิเคราะห์ปัญหา, สถานการณ์ปัจจุบันและความต้องการ งานในขั้นนี้แบ่งออกเป็นงานย่อยหกงานดังนี้ คือ

1. กำหนดปัญหาของระบบงานเดิม (Problem listing)

ก่อนที่จะวิเคราะห์ปัญหาได้นั้น จำเป็นต้องมีการกำหนดปัญหาของระบบงานเดิมให้ได้เสียก่อน ปัจจัยที่จะช่วยในการกำหนดปัญหาได้ก็คือ ประสบการณ์และความเข้าใจในระบบงานเดิม เมื่อกำหนดปัญหาได้แล้วก็ต้องทำการนำเสนอปัญหานั้น โดยจะนำเสนอในรูปแบบของตารางปัญหา (Problem Table) ซึ่งตารางนี้ไม่มีรูปแบบที่แน่นอน แต่โดยทั่ว ๆ ไปจะประกอบไปด้วย

- หมายเลขอ้างอิงปัญหา ที่เริ่มต้นด้วยอักษร 'P' และตามด้วยตัวเลข
- ปัญหา
- คำอธิบายปัญหา เป็นต้น

2. การวิเคราะห์กลุ่มที่สนใจ (Analysis of interest groups)

การวิเคราะห์กลุ่มที่สนใจ เป็นการแบ่งบุคคลที่เกี่ยวข้องกับกิจกรรมขององค์การ และได้รับผลกระทบจากปัญหาที่ได้ในขั้นก่อน ออกเป็นกลุ่ม ๆ ซึ่งกลุ่มที่ได้นี้จะถูกนำเสนอในรูปแบบของ บัญชีรายชื่อกลุ่มที่สนใจ (List of interest group) ที่ประกอบด้วย

- หมายเลขประจำกลุ่ม ที่เริ่มต้นด้วยอักษร 'I' และตามด้วยตัวเลข
- หมายเลขอ้างอิงปัญหา ที่มีผลกระทบต่อกลุ่ม
- หมายเลขอ้างอิงกิจกรรม ที่เกี่ยวข้องกับกลุ่มบุคคลและปัญหา

ในระหว่างที่ทำการวิเคราะห์กลุ่มที่สนใจ อาจจะเกิดกลุ่มที่สนใจกลุ่มใหม่ขึ้นมาพร้อมกับปัญหาใหม่ขึ้นได้ ซึ่งเมื่อเกิดกรณีเช่นนี้ขึ้นแล้วจะต้องทำการปรับปรุงตารางปัญหาเดิมให้ถูกต้องด้วย

3. จัดกลุ่มปัญหา (Problem grouping)

การที่จะทำงานกับปัญหาจำนวนมากในเวลาเดียวกันนั้น เป็นงานที่ยากลำบาก ดังนั้นจึงต้องทำการแบ่งปัญหาในตารางปัญหาออกเป็นกลุ่ม ๆ ซึ่งกลุ่มของปัญหาที่ได้มานี้จะถูกนำเสนอในรูปแบบของ ตารางกลุ่มปัญหา (Problem group table) ที่ประกอบด้วย

- หมายเลขอ้างอิงกลุ่มปัญหา ที่เริ่มต้นด้วย 'G' และตามด้วยหมายเลข
- หมายเลขอ้างอิงปัญหาที่อยู่ในกลุ่มนั้น

4. กำหนดกิจกรรมที่ทำในปัจจุบัน (Description of current activities)

การกำหนดกิจกรรมที่ทำในปัจจุบัน เป็นการแสดงกิจกรรมความสัมพันธ์ในการติดต่อกับปัญหาและกลุ่มที่สนใจ โดยใช้การวิเคราะห์ปัญหาเป็นพื้นฐาน ผลของการทำงานในขั้นนี้คือ โมเดลกิจกรรมของความเหมาะสมในปัจจุบัน

5. กำหนดเป้าหมายของงานที่ทำ (Description of objectives)

โดยปกติกลุ่มสนใจที่แตกต่างกันจะมีความต้องการที่แตกต่างกันออกไป ซึ่งความต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนี้อาจจะขัดแย้งกันก็ได้ ดังนั้นการทำงานในขั้นตอนที่จึงเป็นการสรุปเป้าหมายที่เป็นไปได้มากที่สุด ที่จะสอดคล้องกับความต้องการของทุก กลุ่ม ผลที่ได้จากการทำงานในขั้นนี้ จะถูกบันทึกอยู่ในรูปของ ตารางเป้าหมาย (table of objectives) ที่ประกอบด้วย

- หมายเลขเป้าหมาย ที่แทนด้วยตัวเลข
- เป้าหมาย

6. ประเมินความเหมาะสมในปัจจุบัน (Evaluation of current situation)

ในขั้นนี้เป็นการเปรียบเทียบสิ่งที่ต้องการ (เป้าหมาย) กับสิ่งที่มีอยู่ (ตารางปัญหา และกิจกรรมที่ทำอยู่ในปัจจุบัน) ผลที่ได้เป็นการเปลี่ยนจากปัญหาไปเป็น ความต้องการที่จะเปลี่ยนแปลง (need for changes) ความต้องการที่จะเปลี่ยนแปลงนี้จะถูกบันทึกลงใน ตารางความต้องการที่จะเปลี่ยนแปลง (table of need for changes) โดยแต่ละ ตารางจะสำหรับแต่ละกลุ่มปัญหา ซึ่งตารางนี้จะถูกใช้เป็นพื้นฐานในการศึกษาแนวทางต่างๆ ที่จะใช้เปลี่ยนแปลงต่อไป

2.1.2 ศึกษาแนวทางต่าง ๆที่จะใช้เปลี่ยนแปลง

การศึกษาแนวทางต่าง ๆที่จะใช้เปลี่ยนแปลง จะกระทำกับแต่ละกลุ่มปัญหา โดยใช้วิธีดังต่อไปนี้

1. สร้างแนวทางต่าง ๆที่จะใช้เปลี่ยนแปลง

เป็นการคิดหาแนวทางการ เปลี่ยนแปลงที่เป็นไปได้ทั้งหมด ที่สามารถช่วยแก้ปัญหาที่เกิดขึ้นกับระบบงานได้ ซึ่งแนวทางการเปลี่ยนแปลงนี้ไม่สามารถแนะนำกันได้ ขึ้นอยู่กับประสบการณ์ของแต่ละคน แนวทางที่จะใช้ เปลี่ยนแปลงที่ได้มานี้จะถูกบันทึกไว้ใน ตารางแนวทางที่จะใช้เปลี่ยนแปลง (table of change alternatives)

2. บรรยายแนวทางที่จะใช้เปลี่ยนแปลง

ในขั้นนี้เป็นการวิเคราะห์และประเมินผลที่ได้จากแต่ละแนวทางที่จะใช้เปลี่ยนแปลง โดยผลจะอยู่ในรูปของโมเดลกิจกรรม ที่ประกอบด้วย แผนภูมิกิจกรรม, รายการกำกับ และ ตารางแสดงคุณสมบัติ

3. ประเมินแนวทางที่จะใช้เปลี่ยนแปลง

เป็นการเปรียบเทียบข้อดีและข้อเสียของแนวทางที่จะใช้เปลี่ยนแปลงที่มีต่อ มนุษย์, สังคมและเศรษฐกิจว่าเป็นอย่างไรบ้าง คู่หรือไม่ที่จะทำการเปลี่ยนแปลงนั้น เพื่อเป็นพื้นฐานในการเลือกแนวทางสำหรับเปลี่ยนแปลงต่อไป

2.1.3 เลือกแนวทางสำหรับเปลี่ยนแปลง

งานในส่วนสุดท้ายของการวิเคราะห์เปลี่ยนแปลงก็คือ การเลือกแนวทางสำหรับเปลี่ยนแปลง โดยใช้วิธีดังต่อไปนี้

สผวน .

แผนภูมิกิจกรรม

ผู้วิเคราะห์

กลุ่มพัฒนาระบบสารสนเทศ

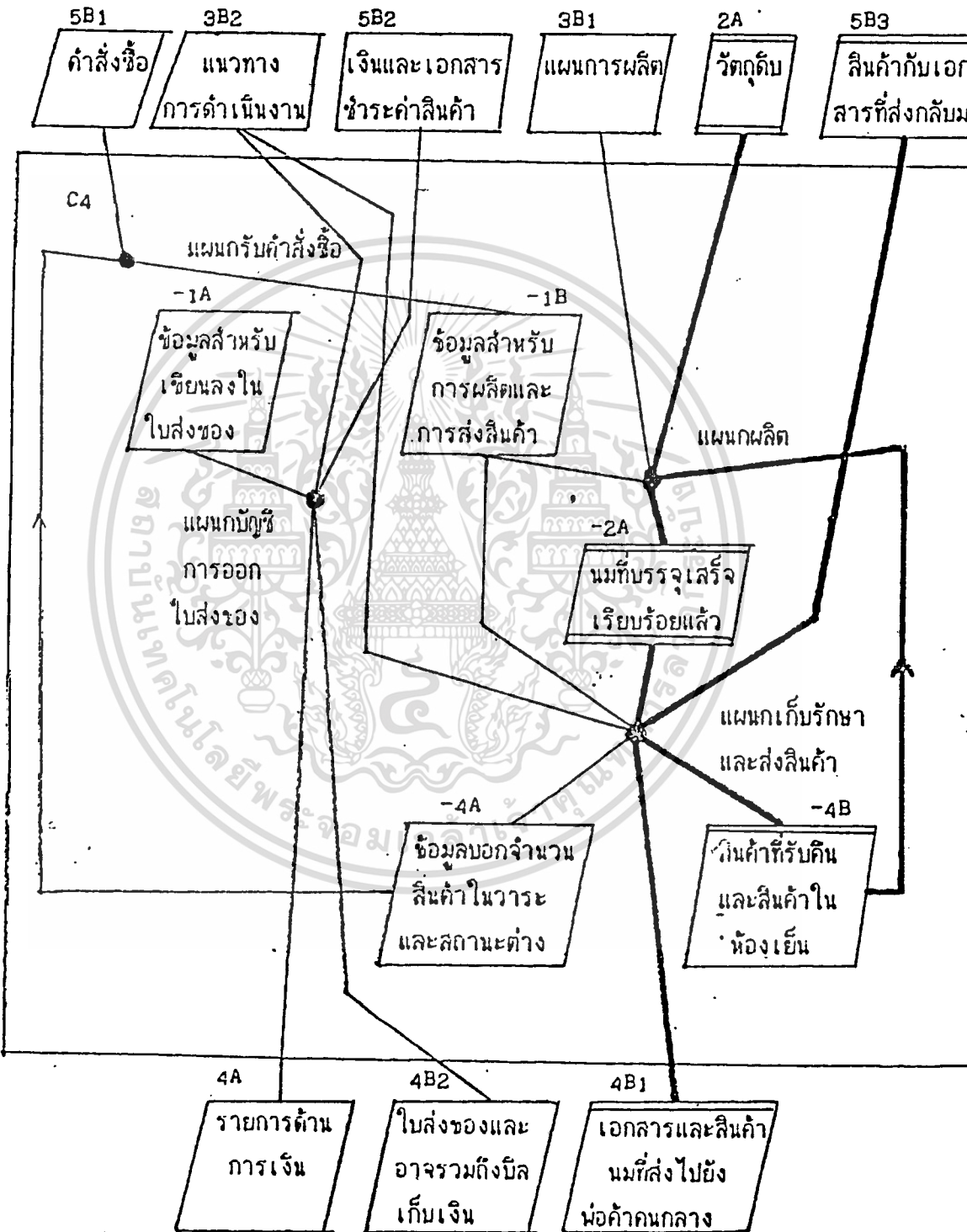
วันที่ : 9 เม.ย. 2533

เลขอ้างอิง C4

ขอบเขต :

ระบบรับคำสั่งซื้อและส่งสินค้า

เวอร์ชัน 4



รูปที่ 5 แผนภูมिरายละเอียดขยายรายละเอียดจากกิจกรรมหมายเลข 4 ของแผนภูมิลรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องแจ้งไปยังเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สผน .

รายงานกำกับ

ผู้วิเคราะห์:

แผนภูมิกิจกรรม

กลุ่มพัฒนาระบบสารสนเทศ

เลขอ้างอิง C4

ขอบเขต :

วันที่ : 9 เม.ย. 2539

ระบบรับคำสั่งซื้อและการส่งสินค้า : ผู้ผลิตนม

เวอร์ชัน 4

2A วัตถุประสงค์

2A1 น้ำนมดิบ

2A2 ส่วนผสม เช่น น้ำตาล เป็นต้น

2A3 บรรจุภัณฑ์

3B1 แผนงาน

3B11 แผนการผลิตระยะยาว

3B12 แผนควบคุมการผลิตระยะสั้น

3B2 แนวทางการดำเนินงาน

3B21 แนวทางการจัดระบบบัญชีและการออกใบส่งของ (ไปยังกิจกรรมหมายเลข 43)

3B22 แนวทางการดำเนินงานสำหรับการเก็บรักษาและการส่งสินค้า (ไปยังกิจกรรมหมายเลข 44)

5B1 คำสั่งซื้อ อาจเป็นใบสั่งซื้อหรือสั่งทางโทรศัพท์

5B11 ใบสั่งซื้อจากลูกค้าโดยตรง

5B12 คำสั่งซื้อทางโทรศัพท์

5B2 เงินชำระค่าสินค้า หลังจากที่พ่อค้าคนกลางได้รับสินค้าแล้ว

5B21 เงินชำระค่าสินค้า

5B22 ใบแสดงหลักฐานการชำระเงิน

5B3 สินค้าที่เจือปน ไม่ตรงตามที่สั่ง ถูกส่งคืนกลับมาพร้อมเอกสาร

5B31 ตัวสินค้าที่ถูกส่งคืนกลับมา

5B32 ใบแสดงหลักฐานการส่งคืน

4 ผู้ผลิตนม (การผลิตและจำหน่ายนมของผู้ผลิตนม)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเฉพาะเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1 แผนกรับคำสั่งซื้อจากลูกค้า
- 1A ข้อมูลการสั่งซื้อสำหรับเขียนลงในใบส่งของ
- 1B ข้อมูลการสั่งซื้อสำหรับการผลิตและการส่งสินค้า เช่น จำนวนสินค้าที่ส่ง สถานที่ส่งสินค้า
- 2 แผนการผลิต
- 2A สินค้ารวมทั้งผ่านการบรรจุเรียบร้อยแล้ว
- 3 แผนบัญชีและใบส่งของ
- 4 แผนเก็บรักษาและส่งสินค้า
- 4A ข้อมูลจำนวนสินค้าในวาระ และสถานะต่าง ๆ คือ สินค้าที่ถูกสั่งซื้อ, ได้รับคืน และ ได้ส่งถึงมือค้า
- 4A1 ใบบันทึกจำนวนสั่งซื้อ
- 4A2 ใบบันทึกจำนวนรับคืน
- 4A3 ใบบันทึกจำนวนที่ได้ส่งถึงมือพ่อค้าคนกลาง
- 4B สินค้าที่รับคืนมาและสินค้าในห้องเย็น
- 4B1 สินค้าที่รับคืนมา
- 4B2 สินค้าในสต็อก (ห้องเย็น)

-
- 4A รายงานการเงินของผู้ผลิตเกี่ยวกับระบบการออกแบบใบส่งของ และบัญชีแยกประเภทต่าง ๆ
 - 4B1 สินค้าสำเร็จรูปที่ส่งไปให้พ่อค้าคนกลางพร้อมเอกสาร
 - 4B2 ใบส่งของและอาจรวมถึงบิลเก็บเงินค่าสินค้าวางมัดก่อน ๆ

รูปที่ 6 รายการกำกับสำหรับกิจกรรมหมายเลข 4

ข้อกำหนดในการเขียนแผนภูมิรายละเอียด มีดังต่อไปนี้

1. เลขที่อ้างอิง

เลขที่อ้างอิงของแผนภูมิกิจกรรมจะประกอบด้วยตัวอักษรย่อหน้าหน้า โดยตัวอักษรย่อหน้าจะใช้อ้างอิงถึงแผนภูมิกิจกรรมทุกภาพในระบบ (และจะตั้งขึ้นด้วยความหมายใดก็ได้ เช่น ชื่อของกลุ่มผู้วิเคราะห์ระบบ เป็นต้น ในกรณีของ สผ.นี้ คือ ตัว C และมีตัวเลขกำกับท้ายสำหรับแผนภูมิสรุป ซึ่งในระบบมีเพียงภาพเดียวจะใช้เลขโดดส่วนแผนภูมิรายละเอียดจะใช้เลขอ้างอิงของกิจกรรมที่นำมาเขียนเป็นแผนภูมิรายละเอียดนั้น เป็นเลขกำกับท้ายจากตัวอย่างเลขที่กำกับท้ายของเลขอ้างอิงคือเลข 4 เพราะมาจากกิจกรรมหมายเลข 4 (ของแผนภูมิรายละเอียด)

2. พื้นที่อยู่นอกกรอบ ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช็ดที่อยู่ภายนอกกรอบสี่เหลี่ยมไม่ว่าด้านบนหรือด้านล่างก็ตาม ได้แก่ บรรดาเช็ดหรือเช็ดย่อยที่ล้วนต้องอ้างอิงถึงมาแล้วจากแผนภูมิกิจกรรม หรือรายการกำกับที่อยู่เหนือแผนภูมิรายละเอียดนี้ (กรณีนี้แผนภูมิกิจกรรมที่ว่าก็คือ แผนภูมิสรุป) จะต้องมีการทุกเช็ดด้วยตามแผนภูมิสรุป CO เมื่อคราวที่แล้วเช็ด 2A, 3B และ 5B เป็นอันขาด ส่วนเช็ด 4A และ 4B เป็นเอาต์พุตของกิจกรรมผู้ผลิตนม ดังนั้นแผนภูมิรายละเอียด C4 ในรูปที่ 3 จะต้องมิเช็ด 2A, 3B และ 5B เป็นอันขาดที่ด้านบนนอกกรอบสี่เหลี่ยม ส่วนเช็ด 4A และ 4B เป็นเอาต์พุตที่ด้านล่างนอกกรอบสี่เหลี่ยม จะมีเช็ดที่เกินมาหรือขาดไปจากนี้ไม่ได้ แต่เนื่องจากว่าเช็ด 3B, 5B และ 4B ได้แยกเป็นเช็ดย่อยไว้ที่รายการกำกับของแผนภูมิสรุป CO อยู่แล้ว จึงสามารถแสดงการแจกแจงแยกเช็ดย่อยเหล่านั้นไปตามกิจกรรมต่าง ๆ ของแผนภูมิรายละเอียด C4 ได้

3. เช็ดที่อยู่ภายนอกกรอบ

เช็ดและกิจกรรมทั้งหลายภายในกรอบสี่เหลี่ยม จะต้องตั้งหมายเลขอ้างอิงของมันด้วยหลักเกณฑ์เดียวกับที่กล่าวไปแล้ว แต่มิได้หมายความว่าแต่ละแผนภูมิกิจกรรมจะมีหมายเลขอ้างอิง 1, 2, ... หรือ 1A, 1B, ... เหมือนหรือซ้ำกัน ขอให้สังเกตเห็นเครื่องหมาย "--" หน้าตัวเลขอ้างอิงเหล่านั้น เครื่องหมายนี้บอกให้รู้ว่า เช็ดหรือกิจกรรมนั้น ๆ ได้ละหมายเลขอ้างอิงที่อยู่ข้างหน้ามันเอาไว้ - หมายเลขนี้ก็คือตัวเลขที่กำกับท้ายเลขอ้างอิงของแผนภูมิกิจกรรม (แผนภูมิรายละเอียด) ซึ่งคือหมายเลขตัวเดียวกับหมายเลขอ้างอิงของกิจกรรมจากแผนภูมิกิจกรรมที่อยู่เหนือแผนภูมิกิจกรรมนี้ซึ่งเรานำมาขยายเป็นแผนภูมิกิจกรรมนี้

จากตัวอย่างหมายเลขที่ละไว้ในตัวอย่างก็คือ เลข 4 ซึ่งมาจาก C4 ของเลขอ้างอิงแผนภูมิกิจกรรม หรือมาจากกิจกรรมหมายเลข 4 "ผู้ผลิตนม" ของแผนภูมิสรุป CO ดังนั้นเวลาจะอ้างถึงเช็ดหรือกิจกรรมโดยไม่ให้สับสนแล้ว ก็ต้องอ้างอิงให้ชัดเจน เช่น จะอ้างถึงกิจกรรม "แผนกเก็บรักษาและส่งสินค้า" ก็ต้องบอกว่า กิจกรรมหมายเลข 4 ของแผนภูมิกิจกรรม C4 หรือเรียกอย่างสั้น ๆ ว่า กิจกรรมหมายเลข 44 (ตัวอย่างในรายการกำกับ ซึ่งอธิบายความหมายของเช็ดย่อย 3B22)

4. จำนวนกิจกรรม

จำนวนกิจกรรมย่อยในแต่ละแผนภูมิกิจกรรมจะต้องไม่เกินเท่ากับกิจกรรม มิฉะนั้นหากมีการอ้างอิงถึงกิจกรรมที่สิ้นการใช้ตัวเลขอ้างอิงอาจทำให้เกิดความสับสน

5. เส้นการเคลื่อนที่ในกราฟ

ห้ามเส้นแสดงการเคลื่อนที่ของเช็ดเชื่อมต่อระหว่างกิจกรรมหนึ่งไปยังกิจกรรมหนึ่งหรือเช็ดหนึ่งไปยังเช็ดหนึ่งโดยตรง ต้องเชื่อมจากกิจกรรมไปยังเช็ด หรือจากเช็ดไปยังกิจกรรมเท่านั้น

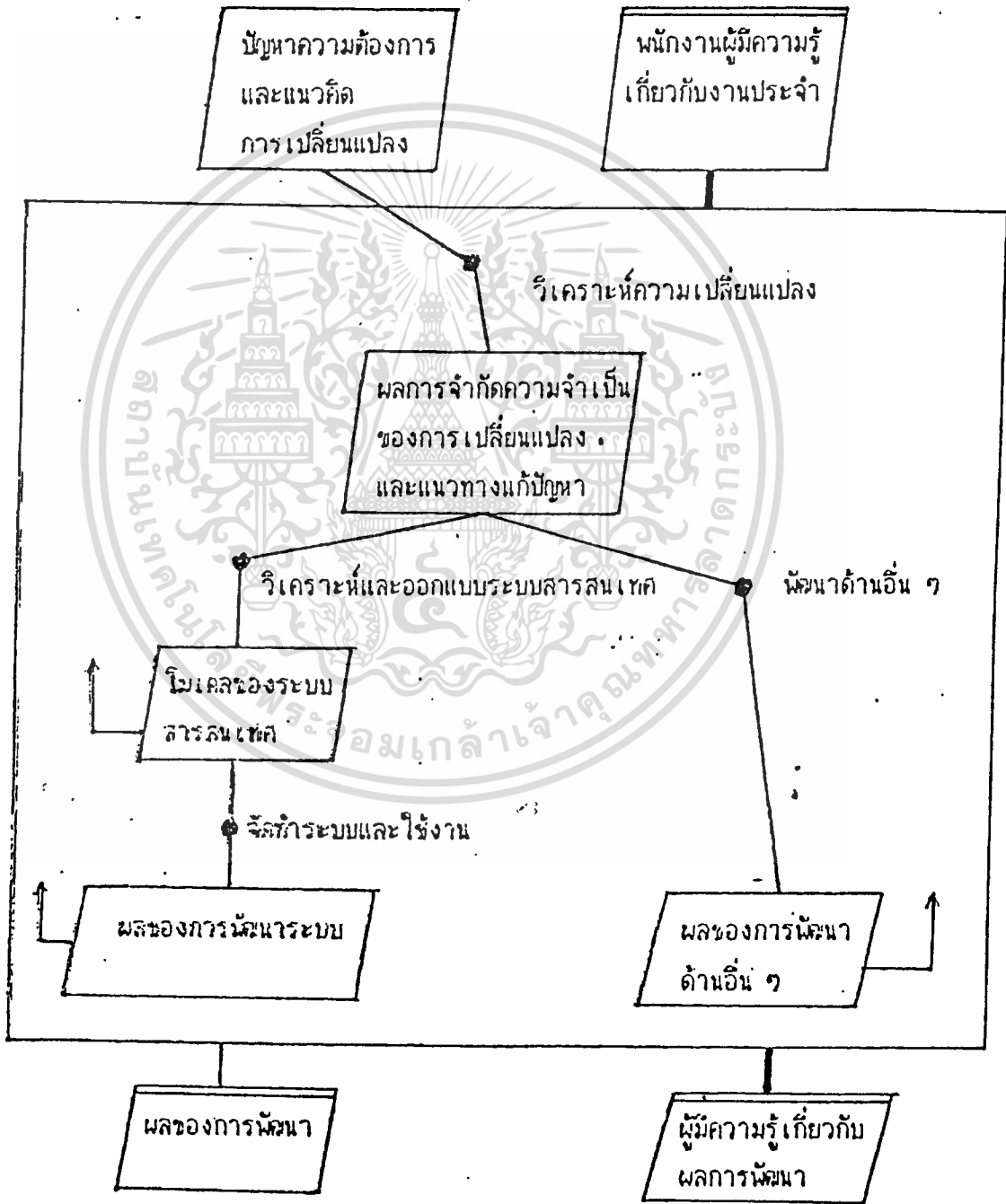
6. ตำแหน่งของแผนภูมิกิจกรรม

ในการวางตำแหน่งของแผนภูมิกิจกรรมและรายการกำกับในสมุดหรือแฟ้มเอกสารนั้น เอกสารนี้เป็นเอกสารที่ส่งวันเวลาสำหรับการเขียนเพื่อไว้ศึกษาเท่านั้น เมื่ออยู่ในแฟ้มไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตามธรรมเนียมที่ยึดถือกันจะจัดให้รายการกำกับอยู่ที่หน้ากระดาษทางซ้ายมือและแผนภูมิกิจกรรมอยู่หน้าทางขวามือ

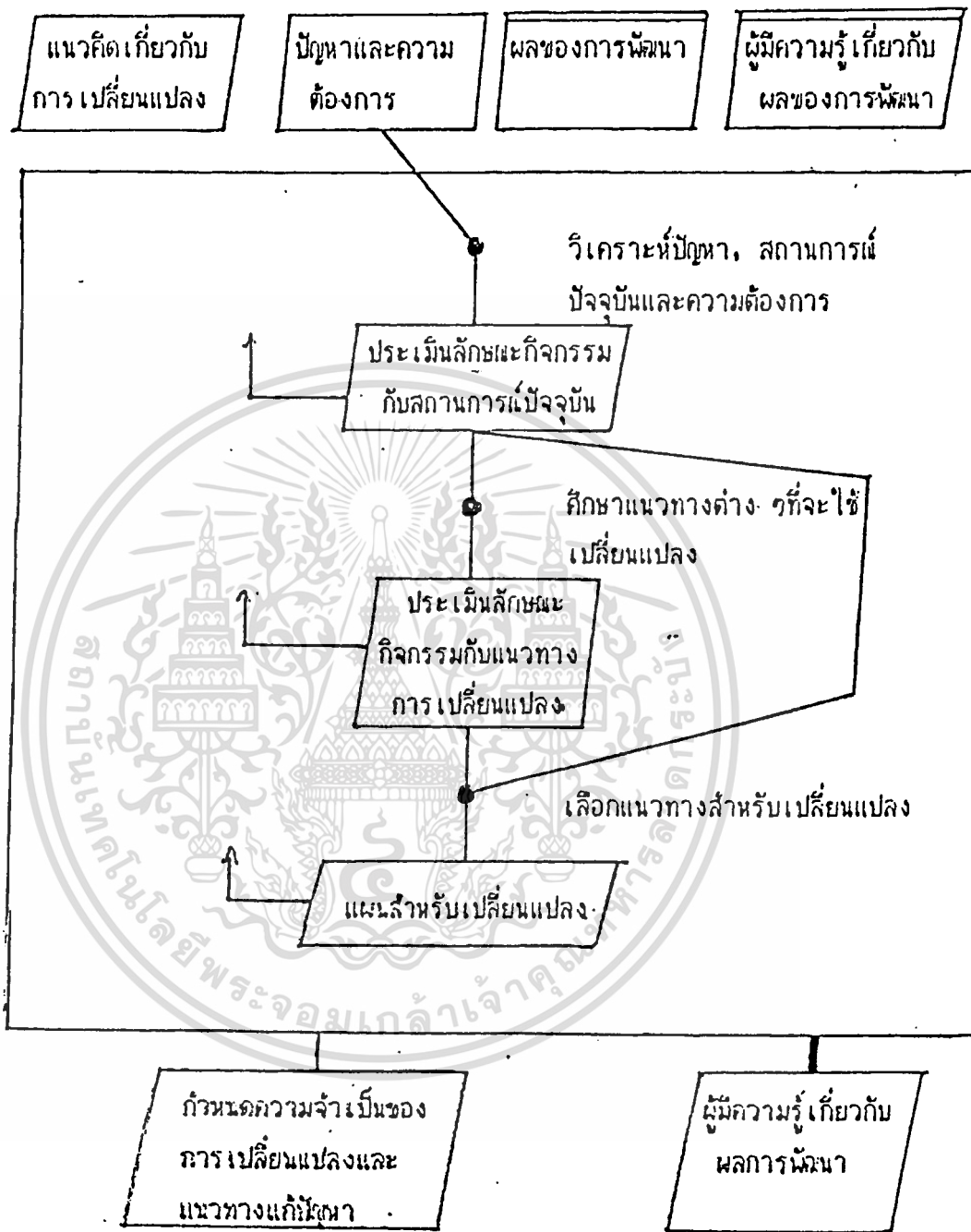
2. การวิเคราะห์และออกแบบงาน

จากรูปที่ 1 ข้างต้นเราสามารถขยายรายละเอียดของกิจกรรมการพัฒนาระบบการต่อไปได้อีก ดังแสดงในรูปที่ 7



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้รูปที่ 7 รายละเอียดเกี่ยวกับการพัฒนาระบบการนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ออกในแต่ละกิจกรรมได้ ซึ่งรายละเอียดเหล่านี้สามารถแสดงได้โดยตารางแสดงคุณสมบัติ

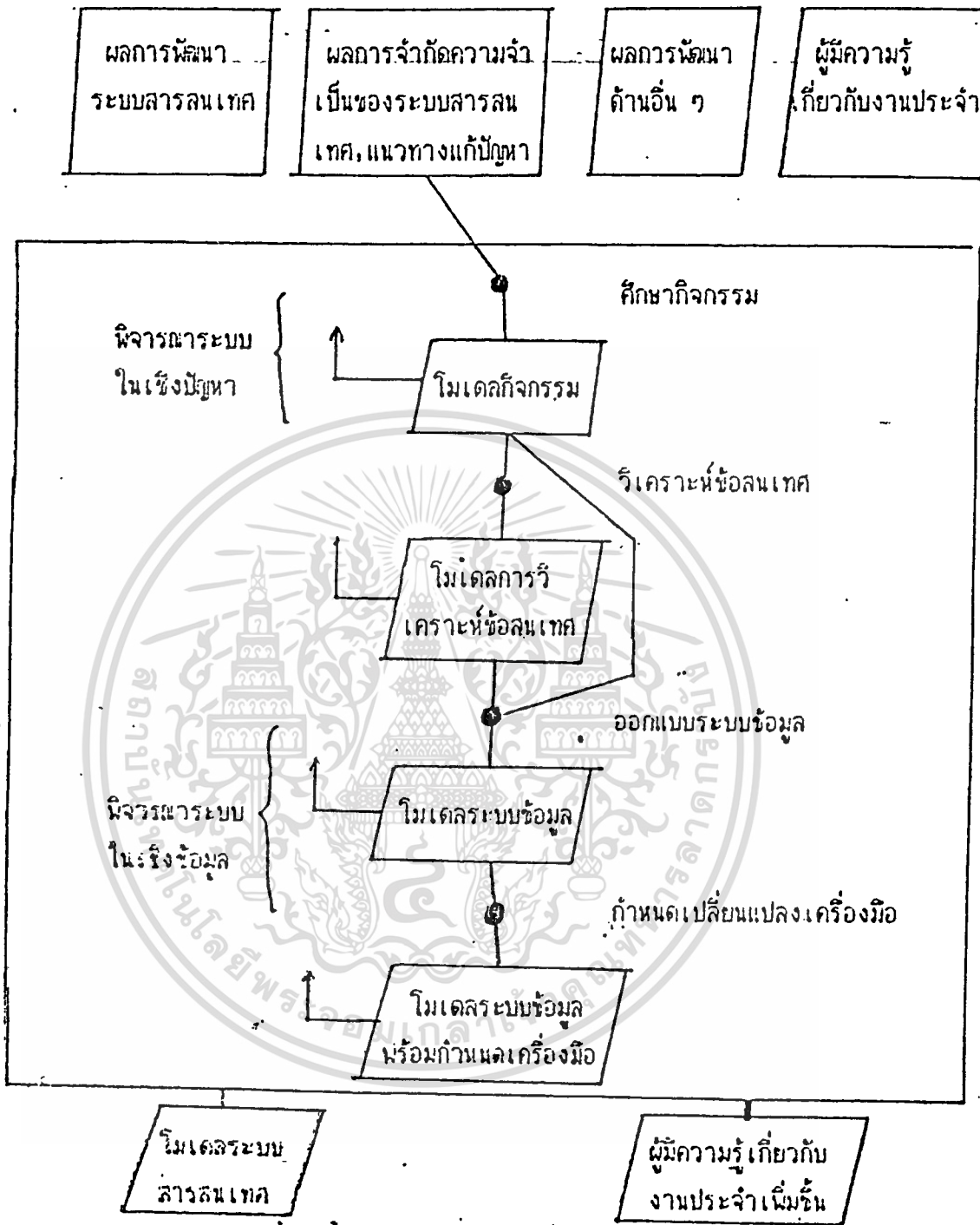


รูปที่ 8 วิธีการวิเคราะห์การเปลี่ยนแปลง

จากรูปที่ 8 จะได้ว่าวิธีที่ใช้ในการวิเคราะห์การเปลี่ยนแปลงมีอยู่ด้วยกันสามขั้นตอน โดยเรียงตามลำดับดังนี้ คือ

- วิเคราะห์ปัญหา, สถานการณ์ปัจจุบันและความต้องการ
- ศึกษาแนวทางต่าง ๆ ที่จะใช้เปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนเลือกแนวทางสำหรับเปลี่ยนแปลงเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ๑ ขั้นตอนในการวิเคราะห์และออกแบบระบบ

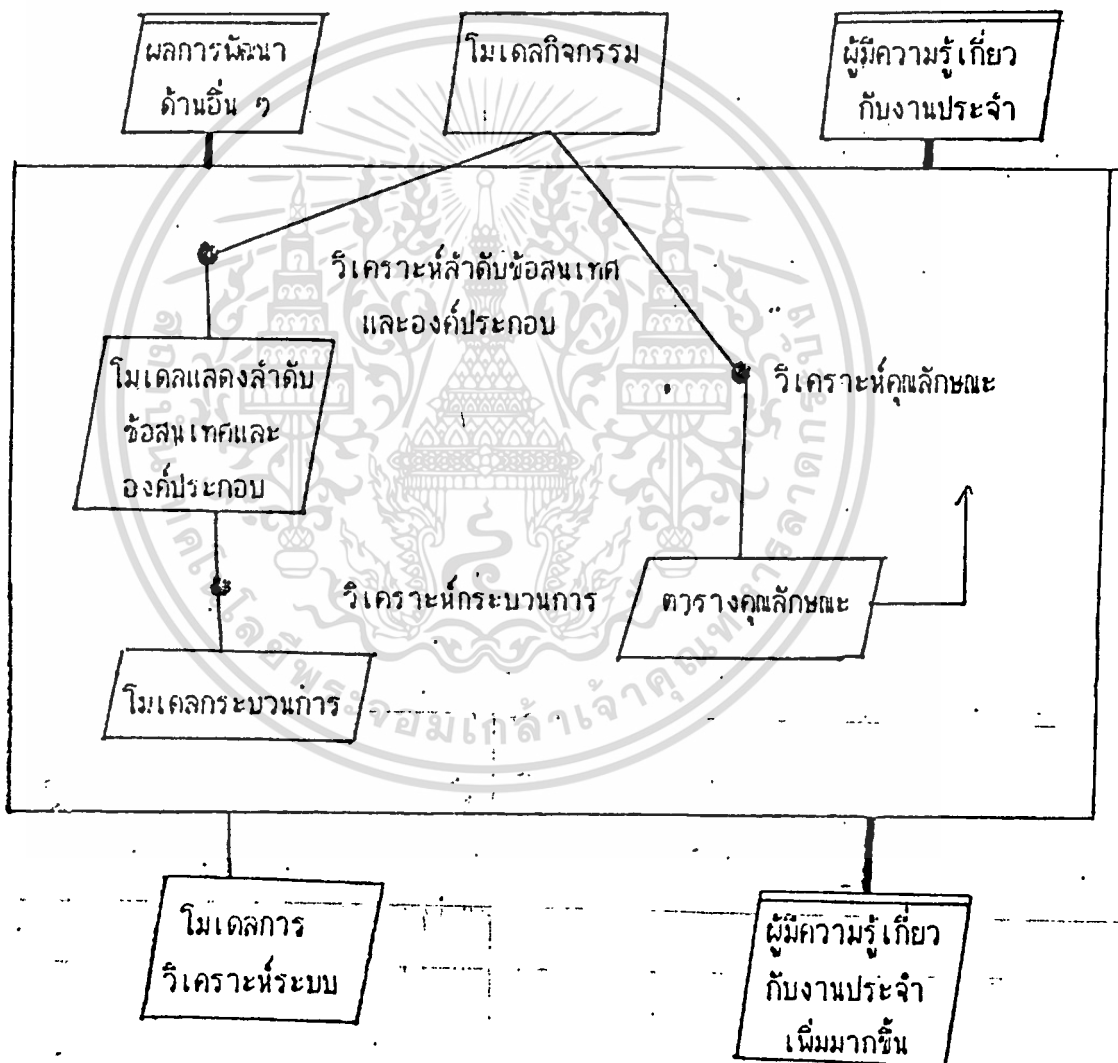
การศึกษากิจกรรมและระบบงานช่วยของงานปัจจุบัน จะเน้นความสำคัญที่ผู้ใช้ เนื่องจากผู้ใช้ระบบ คือผู้ที่รู้งานดีทีสุด รู้ว่าปัญหาเกิดขึ้นที่ใด ทั้งยังเป็นผู้ที่ต้องรับผิดชอบและทำงานในระบบใหม่ต่อไปอีกด้วย ถ้าหากผู้ใช้ไม่ให้ความร่วมมือในการศึกษาระบบแล้ว ก็ยากที่งานนี้จะประสบผลสำเร็จได้

งานในขั้นนี้เป็นเพียงการทำความเข้าใจระบบงาน ดังนั้นจึงไม่ควรด่วนตัดสินใจว่า เอกสารนี้เป็นเอกสารที่ล้าสมัยแล้ว หรือแก้ไขในเพื่อปรับปรุงแก้ไขในขั้นต่อไป อย่างไรก็ตามไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบย่อยที่ได้ทำการจำแนกไว้แล้ว ควรมีการทำงานอย่างไร

2.2.2) การวิเคราะห์ข้อสนเทศ

มีจุดมุ่งหมายเพื่อที่จะพิจารณาว่า ระบบที่จะสร้างขึ้นใหม่นั้นควรจะทำอะไรได้บ้าง และควรมีขอบเขตสักแค่ไหน งานวิเคราะห์นี้ก่อให้เกิดประโยชน์สองประการ ประการแรกคือ ใช้เป็นเครื่องมือสำหรับสื่อสารระหว่างกลุ่มต่าง ๆ ที่มีส่วนเกี่ยวข้องกับระบบงานที่ทำ และประการที่สองคือ ใช้เป็นพื้นฐานสำหรับการออกแบบระบบข้อมูล งานในการวิเคราะห์ข้อสนเทศมีลักษณะดังแสดงในรูปที่ 10



รูปที่ 10 ขั้นตอนในการวิเคราะห์ข้อสนเทศ

ในการวิเคราะห์จะต้องกระทำกับงานย่อยของทุกระบบที่เกี่ยวข้อง ซึ่งการวิเคราะห์จะต้องพิจารณาละเอียดถี่ถ้วนโดยขึ้นอยู่กับว่า ระบบงานย่อยนั้นจะถูกเปลี่ยนให้เป็นงานที่นำคอมพิวเตอร์เข้ามาช่วยหรือไม่ ถ้าระบบใดยังไม่มีการเปลี่ยนแปลงก็ให้ทำการวิเคราะห์

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพียงคร่าวๆ เพียงเพื่อให้เข้าใจกระบวนการทำงานที่พอแล้ว ส่วนระบบย่อยใดที่จะเปลี่ยน ก็จำเป็นที่จะต้องวิเคราะห์กันอย่างละเอียดก็เท่านั้น

การวิเคราะห์ข้อสนเทศ เริ่มด้วยการเขียนแผนภาพแสดงลำดับขั้นตอนการสร้างและ การใช้ข้อสนเทศ ซึ่งเรียกว่า Information Precedence Graph (I-Graph) (แผนภาพนี้มีลักษณะคล้ายกับแผนภูมิกิจกรรม) ซึ่งงานการวิเคราะห์ข้อสนเทศถูกแบ่งออกเป็นสองขั้นตอนคือ ส่วนแรก เป็นการวิเคราะห์ข้อสนเทศและองค์ประกอบ งานอีกส่วน เป็นการวิเคราะห์กระบวนการ

การทำงานเริ่มจากการนำแผนภูมิกิจกรรมมาย่อว่า ใช้ข้อมูลและข้อสนเทศอะไรบ้าง แล้วจึงสาวต่อไปเรื่อย ๆ จนได้กลุ่มข้อมูลที่ เป็น อินพุต จากนั้นจึงทำการวิเคราะห์ ส่วนประกอบของข้อมูลว่า มีรายละเอียดอะไรบ้าง และแต่ละส่วนสัมพันธ์กันอย่างไร แล้วจึงจัดทำแผนภาพที่เรียกว่า Component-relation-graph หรือ C-Graph เมื่อรู้รายละเอียดข้อมูลแล้ว ก็ทำการวิเคราะห์กระบวนการทำงานว่า ระบบย่อยนั้นๆทำงานอย่างไร มีขั้นตอนเป็นเช่นไร และทำการจัดมันทีกเอาไว้

การที่เราจำเป็นต้องศึกษาวิเคราะห์หาความสัมพันธ์ของข้อมูลและข้อสนเทศ ก่อนที่จะทำการวิเคราะห์กระบวนการทำงานนั้น เนื่องจากวิธีนี้ช่วยให้สามารถกำหนดกลุ่มข้อมูลได้ชัดเจน โดยไม่ต้องกังวลกับกระบวนการ หรืองานต่างๆ ที่จะต้องทำกับข้อมูลนั้น

2.2.3) ออกแบบระบบข้อมูล

มีจุดมุ่งหมาย เพื่อที่จะออกแบบระบบข้อมูล ซึ่งเป็นอิสระ ไม่ขึ้นอยู่กับคอมพิวเตอร์ประเภทใด ว่าจะใช้ ส่วนรับระบบข้อมูลที่ได้จัดทำขึ้นในขั้นก่อน แต่ในความเป็นอิสระนี้จำเป็นจะต้องกำหนดแนวทางให้ดีกว่า จะเป็นระบบคอมพิวเตอร์แบบใด และมีการทำงานอยู่ในโหมดใด เป็นต้น เมื่อกำหนดแนวทางได้แล้วก็เริ่มกำหนดโครงสร้างของข้อมูล หลังจากนี้จะเห็นการนำกระบวนการต่าง ๆ ที่ได้วิเคราะห์ไว้แล้วมาจัดรวมเป็นกลุ่ม แล้วออกแบบโปรแกรมที่จำเป็นขึ้น ดังแสดงในรูปที่ 11

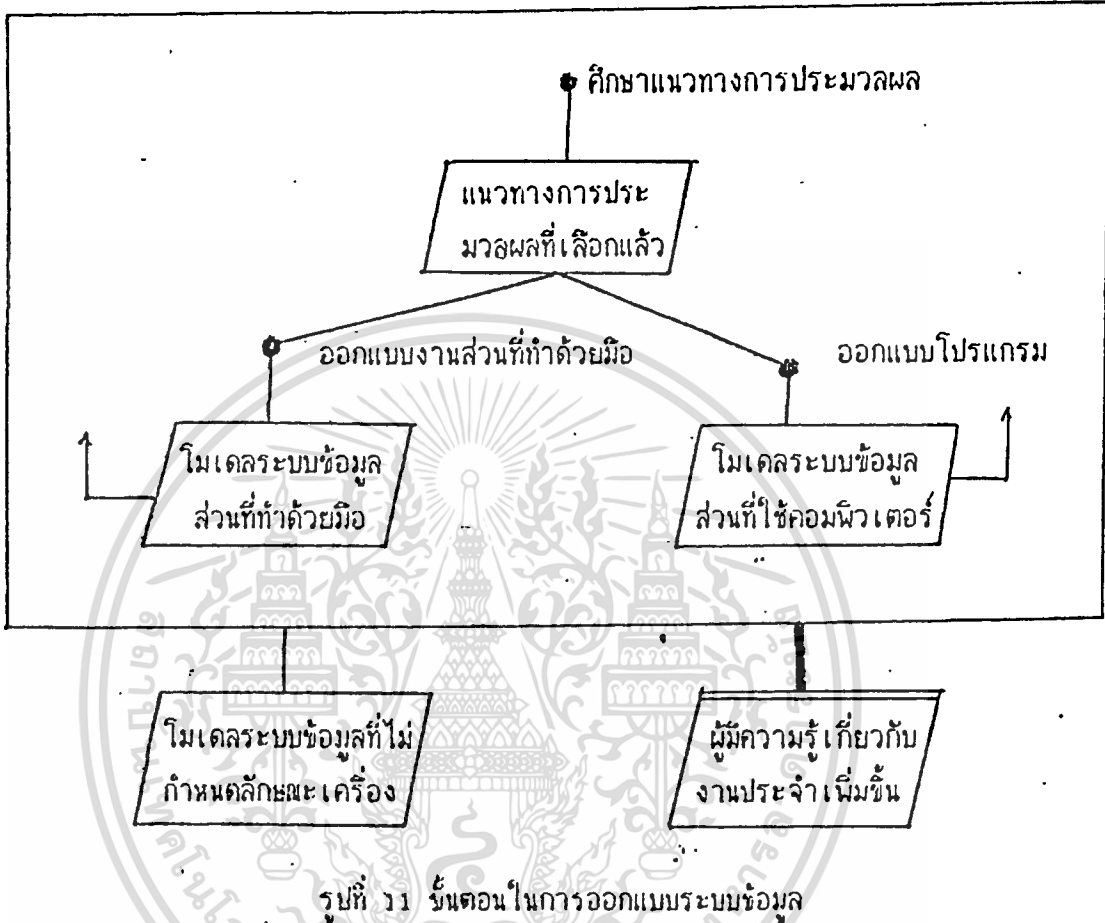
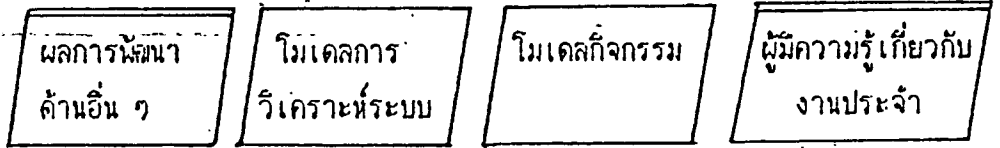
2.2.4) กำหนดเครื่องมือที่เลือกเครื่องมือให้เหมาะสม

มีจุดมุ่งหมาย เพื่อที่จะเลือกเครื่องคอมพิวเตอร์หรืออุปกรณ์ที่จะนำมาใช้ในงานต่างๆ ของระบบที่กำลังออกแบบ การเลือกนี้จะต้องพิจารณาระบบข้อมูลที่จัดทำขึ้นในขั้นก่อน รายละเอียดของการเลือกเครื่องมือ ได้แสดงไว้ในรูปที่ 12

หลังจากทำการวิเคราะห์และออกแบบระบบแล้วจะได้ Information System Model ซึ่งก็คือ นิยามเชิงยาที่จะใช้สร้างระบบนั่นเอง ถ้าหากโมเดลที่ได้ยังไม่ดีพอ ก็ต้องย้อนกลับไปทำการวิเคราะห์และออกแบบระบบใหม่ แต่ถ้าดีแล้วก็จะไปสู่ขั้นตอนการจัดสร้างระบบต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

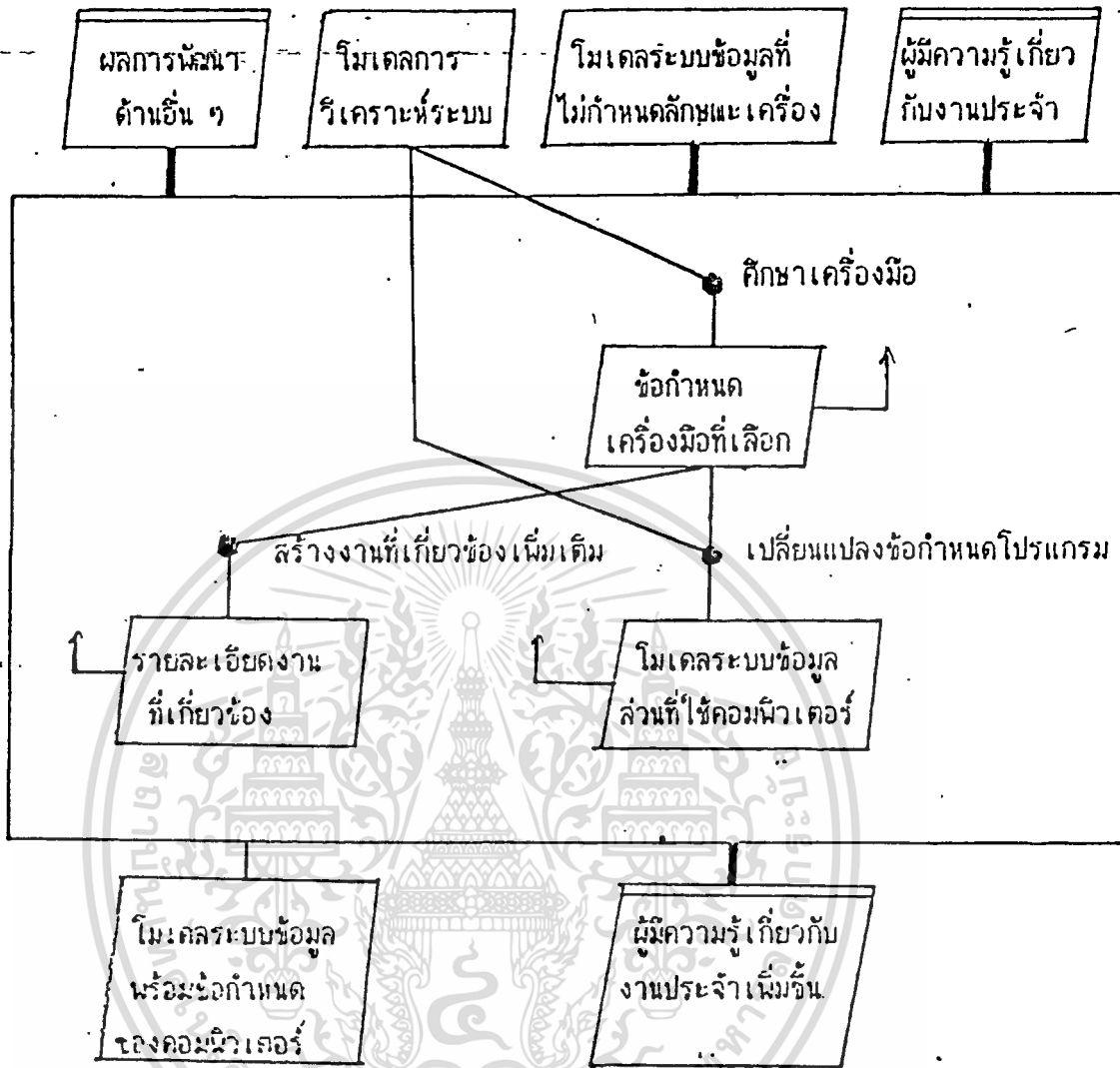


2.3 การจัดสร้างระบบและใช้งาน

การสร้างระบบ เป็นขั้นตอนที่จัดทำตามนิมน์เขียวของระบบที่จัดทำขึ้นตามแนว ISAC จึงประกอบด้วยขั้นตอนสำคัญห้าขั้นตอน คือ

- การเขียนหรือสร้างโปรแกรม
- การจัดทำแผนข้อมูล
- การออกแบบงานที่ทำด้วยมือ
- การทดสอบระบบ
- การสร้างคู่มือ

เมื่อสร้างโปรแกรมและทดสอบระบบเสร็จแล้ว ก็มาถึงขั้นการใช้งานจริง ซึ่งในขั้นนี้ จะต้องมีการเตรียมตัวในด้านต่าง ๆ อันได้แก่



รูปที่ 12 ขั้นตอนการกำหนดเครื่องมือ

- การจัดเตรียมบุคลากร ตลอดจนการฝึกอบรม
- การจัดนิเทศแบบฟอร์มต่าง ๆ
- การจัดทำและเปลี่ยนแปลงแฟ้มข้อมูลเข้าสู่ระบบใหม่
- การใช้งานระบบ

แม้ว่าถึงตอนนี้ระบบใหม่จะทำงานได้ แต่ผู้ที่เกี่ยวข้องกับการสร้างระบบควรจะต้องศึกษาผลลัพธ์ของระบบที่สร้าง เพื่อให้ได้ความรู้และประสบการณ์สำหรับเป็นแนวทางในการพัฒนาระบบอื่น ๆ ต่อไป

3. บทสรุป

การออกแบบระบบโดยวิธี ISAC เป็นการวิเคราะห์และพัฒนาระบบงานใด ๆ โดยมีจุดมุ่งหมายที่จะนำคอมพิวเตอร์ไปช่วยปรับปรุงการทำงาน แต่ก่อนที่จะลงมือสร้างระบบ

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันวิจัยและพัฒนาเทคโนโลยีสารสนเทศแห่งชาติ (สวทช.) ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในจำเป็นต้องทำความเข้าใจกับกิจกรรมขององค์การที่มันอยู่ในปัจจุบัน และพยายามกำหนดปัญหาของระบบงานเดิมและความต้องการให้ได้เสียก่อน * หลังจากนั้นจึงทำตามขั้นตอน โดยเรียงตามลำดับดังนี้ คือ

- การวิเคราะห์ความเปลี่ยนแปลง
- การวิเคราะห์และออกแบบระบบ
- การจัดสร้างระบบและใช้งาน

หลังจากเสร็จสิ้นทุกขั้นตอนข้างต้นแล้ว ก็จะได้ระบบใหม่ที่พร้อมจะใช้งาน แต่การออกแบบยังไม่สิ้นสุด เนื่องจากผู้เกี่ยวข้องกับการสร้างระบบ ควรจะต้องติดตามผลการดำเนินงานของระบบใหม่ เพื่อให้ได้ความรู้และประสบการณ์ในการพัฒนาระบบอื่น ๆ ต่อไป



2.2 แบบแผนความคิดความเข้าใจรวมยอดของวิธีวิเคราะห์ข้อมูลของนิสเฟน
(NIAM Conceptual Schema)

1. บทนำ

ในปี พ.ศ. 2525 องค์การมาตรฐานระหว่างประเทศได้เสนอสถาปัตยกรรมมาตรฐานสำหรับระบบฐานข้อมูลไว้ 3 ระดับ (ดูรูปที่ ๒.1) คือ

1.1 ระดับความคิดความเข้าใจรวมยอด (Conceptual Schema)

ที่ระดับนี้มีบรรทัดฐานสำหรับไวยากรณ์ที่อธิบายยูนิเวิร์สของดิสคอร์ด (Universe of Discourse : UOD) โศกหรืองานที่นำไปประยุกต์ใช้

1.2 แบบแผนภายนอก (External Schema)

แบบแผนนี้อธิบายถึงลักษณะการมองเห็นเฉพาะบางส่วนของแบบแผนความคิดความเข้าใจรวมยอดสำหรับผู้ใช้

1.3 แบบแผนภายใน (Internal Schema)

เป็นแบบแผนที่อธิบายถึงวิธีการจัดการกับกลุ่มของแฟกส์ (Facts) ไว้ในหน่วยความจำภายในและวิธีการเข้าถึงแฟกส์เหล่านี้



รูปที่ ๒.1 สถาปัตยกรรมของระบบฐานข้อมูลที่องค์การมาตรฐานระหว่างประเทศเสนอ

แบบแผนความคิดความเข้าใจรวมยอด ต้องมีคุณสมบัติดังต่อไปนี้

- 1) เป็นอิสระจากแบบแผนภายใน
- 2) สมบูรณ์ในตัว กล่าวคือ มีโครงสร้างเพียงพอที่จะอธิบายยูโอไอได้อย่างสมบูรณ์
- 3) เป็นธรรมชาติ คือ ในการทำแบบจำลองความคิดความเข้าใจรวมยอดนั้นข่าวสารที่พบในยูโอไอพร้อมที่จะใช้ในแบบแผนความคิดความเข้าใจรวมยอด
- 4) วิธีสร้างแบบจำลองต้องธรรมดา และง่ายต่อการเรียนรู้ คุณสมบัติข้อนี้สำคัญที่สุด เพราะผู้ใช้และผู้ที่ไม่ใช่ผู้ออกแบบระบบฐานข้อมูลมักจะเป็นผู้ที่รู้ยูโอไอได้ดีที่สุด ซึ่งบุคคลเหล่านี้ควรสามารถสร้างแบบจำลองความคิดความเข้าใจรวมยอดได้

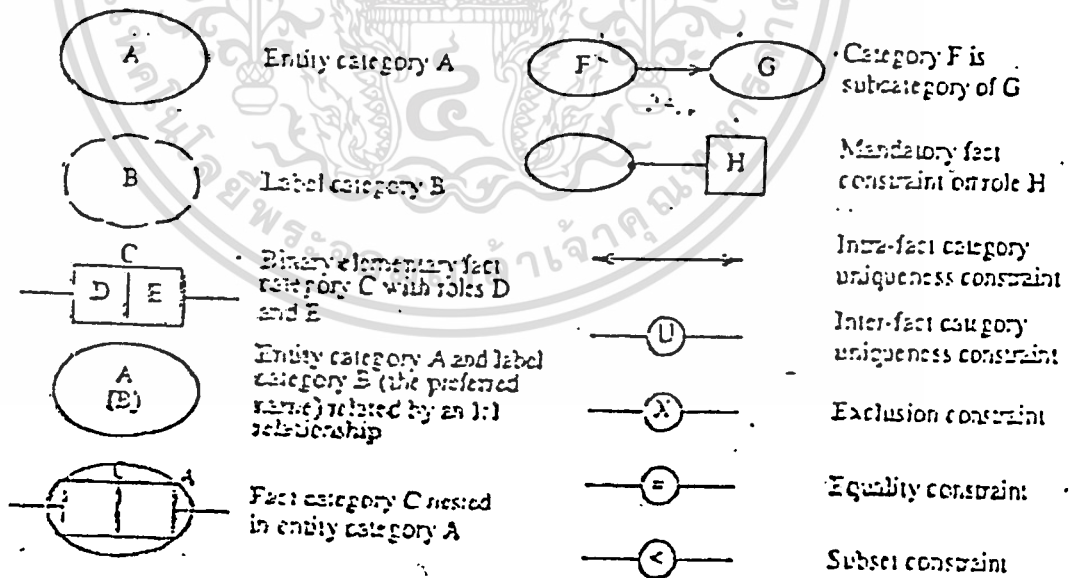
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากยูอิตี ถูกอธิบายในรูปของ แบบแผนความคิดความเข้าใจรวมยอดจึงจำเป็นต้องมีเครื่องมือในการ เปลี่ยนแบบแผนภายใน เพื่อให้ได้ผลผลิตของระบบฐานข้อมูล เครื่องมือนี้เรียกว่า ทรานส์ฟอร์มเมชัน (Transformation) ทรานส์ฟอร์มเมชันนี้ต้องได้รับการนิยามไว้เป็นอย่างดีโดยที่ความหมายต่างๆ ไม่สูญหายในขณะทำการโปรเซส (Process)

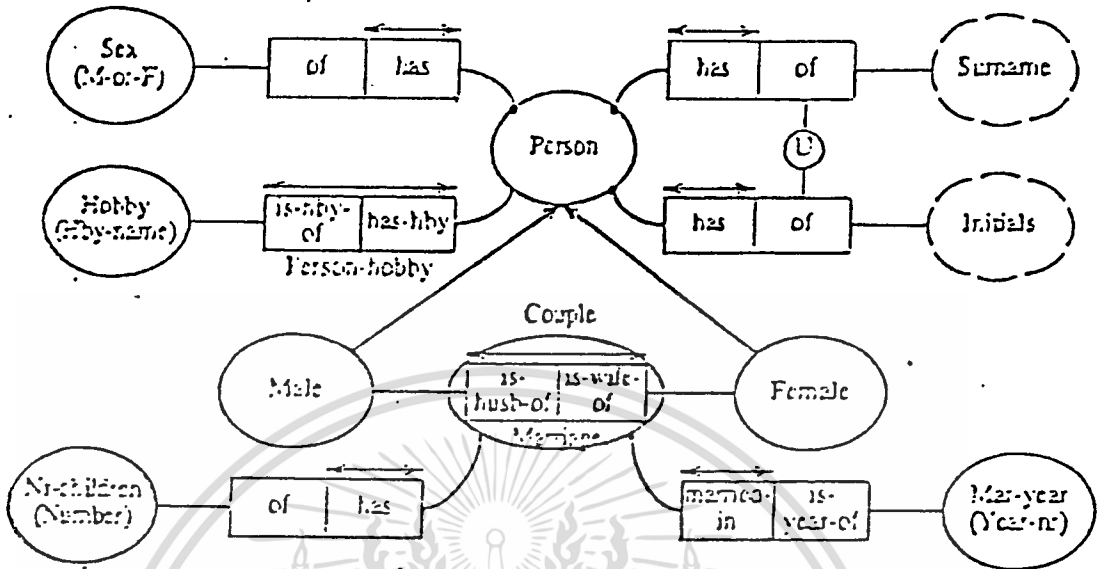
2. แบบจำลองความคิดความเข้าใจรวมยอดของวิธีวิเคราะห์ข้อมูลของนิสเซน (NIAM Conceptual Model)

แบบจำลองความคิดความเข้าใจรวมยอดของวิธีวิเคราะห์ข้อมูลของนิสเซน เป็นแบบจำลองความคิดความเข้าใจรวมยอด ของระบบฐานข้อมูลวิธีหนึ่ง ซึ่งประกอบด้วย สิ่งที่เราสนใจ (Entity Type) เช่น ชื่อคน เพศ งานอดิเรก, สิ่งที่ใช้อ้างอิงสิ่งที่เราสนใจ (Label Type) เช่น นามสกุล, ความสัมพันธ์ระหว่างสิ่งต่างๆ (Facts) ซึ่งจะอยู่ในรูปของ ประธาน กริยากรรม นอกจากนี้ยังมีข้อกำหนดต่างๆ คือ ข้อกำหนดของการเท่ากัน (Equality Constraint), ข้อกำหนดของความเป็นหนึ่ง (Uniqueness Constraint), ข้อกำหนดของความเป็นสับเซต (Subset Constraint), ข้อกำหนดช่วงความถี่ (Frequency Range Constraint), ข้อกำหนดในการที่เมื่อปรากฏข้อมูลอย่างหนึ่งต้องมีข้อมูลอีกอย่างหนึ่งด้วย (Total Role Constraint or Mandatory Role Position Constraint), สัญลักษณ์ของสิ่งต่างๆ นี้แสดงในรูปที่

ข.2



รูปที่ ข.2 แนวความคิดที่เป็นรูปภาพของแบบแผนความคิดความเข้าใจรวมยอดของวิธีวิเคราะห์ข้อมูลของนิสเซน



รูปที่ ๓.๓ ตัวอย่าง

สำหรับข้อกำหนดของความเป็นหนึ่งยังแบ่งได้ไปอีก 2 อย่าง คือ

1) ข้อกำหนดของการปรากฏเพียงครั้งเดียวแบบภายใน (Intra-Fact Category Uniqueness Constraint) ใช้กำหนดจำนวนโรรุ (role) ที่น้อยที่สุดสำหรับแพลตฟอร์มหนึ่งๆ ที่จะทำให้ข้อมูลแต่ละแถวของแพล็ลนั้นไม่ซ้ำกัน

2) ข้อกำหนดของการปรากฏเพียงครั้งเดียวแบบภายนอก (Inter-Fact Category Uniqueness Constraint) ใช้อ้างถึงสิ่งที่เราสนใจมาประกอบกัน เช่น บุคคลถูกอ้างถึงโดยการรวมชื่อกับนามสกุลเข้าด้วยกัน

สำหรับข้อกำหนดแมนเดทอริอริบายได้ดังนี้ เมื่อโรรุใดมีสัญลักษณ์แมนเดทอริปรากฎอยู่ โดยถ้าเรารู้ค่าข้อมูลของสิ่งที่สนใจแล้วค่าของข้อมูลของสิ่งอื่นที่มาเกี่ยวข้องกับสิ่งที่สนใจที่มีสัญลักษณ์แมนเดทอรินั้นต้องมีค่าของข้อมูล เช่น จากรูปที่ ๓.๓ เมื่อปรากฏชื่อของบุคคลหนึ่งบุคคลนั้นต้องมีนามสกุลปรากฏอยู่ด้วย

นิยามที่ 1 : สิ่งที่เราสนใจอาจถูกอ้างถึงแบบยูอิดีได้โดย

ก) เลเบิ้ลที่มีโรรุมาเชื่อมต่อกับโรรุของสิ่งที่เราสนใจแบบ 1:1

ข) กลุ่มของสิ่งที่ใช้อ้างถึงแบบปรุมมุขิของสิ่งที่เราสนใจโดยเป็นสิ่งที่มาเกี่ยวข้องกับแบบร่างแหที่ถกนิยามบนแพล็ลนั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนก) กลุ่มของสิ่งที่ใช้อ้างถึงแบบปรุมมุขิของสิ่งที่เราสนใจโดยเป็นสิ่งที่มาเกี่ยวข้องกับ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กันอันเกิดจากความสัมพันธ์กันระหว่างแฟลคส์

ถ้าสิ่งที่เราสนใจถูกอ้างถึงได้มากกว่า -1 แบบจากข้างบนเราจะกล่าวว่าสิ่งที่เราสนใจนั้นมีซินโนนิม (Synonyms)

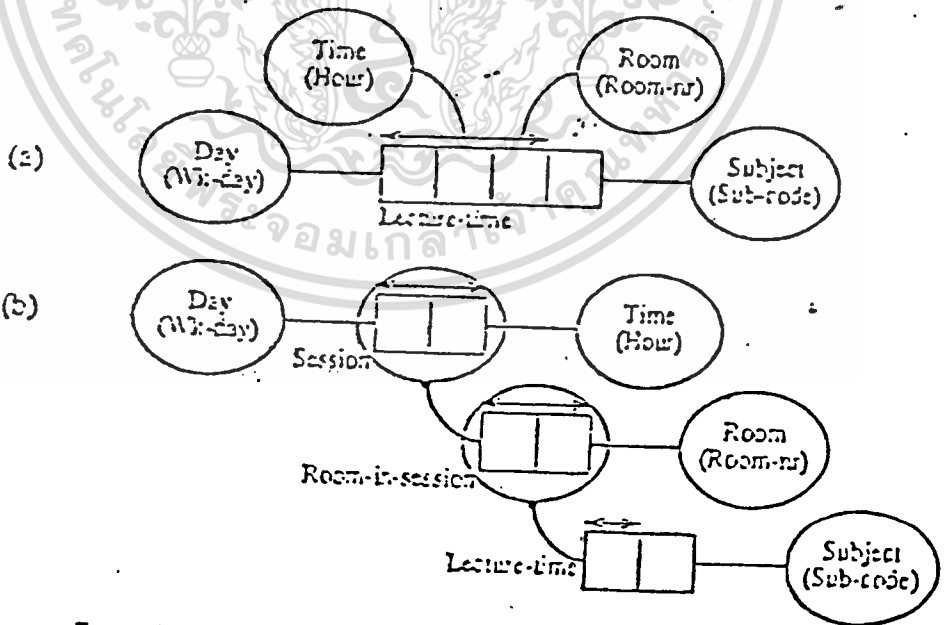
นิยามที่ 2 :

สิ่งที่ใช้อ้างถึงสิ่งที่เราสนใจแบบปฐมภูมิเป็นสิ่งที่ใช้อ้างถึงเพียงอย่างเดียวถ้าไม่มีซินโนนิม แต่ถ้ามีซินโนนิมจะถือว่าสิ่งที่เกิดขึ้นก่อนในนิยามที่ 1 เป็นสิ่งที่ใช้อ้างถึงสิ่งที่เราสนใจแบบปฐมภูมิ

3. คุณสมบัติบางประการ ของแบบแผนความคิดความเข้าใจรวบยอดของวิธีวิเคราะห์ข้อมูลของนิสเซน

3.1 แต่ละแฟลคส์ที่มีโรลจำนวน n (n -ary, $n \geq 2$) จะมีข้อกำหนดความเป็นหนึ่งนิยามบนแฟลคส์นั้นอย่างน้อยหนึ่งโรล

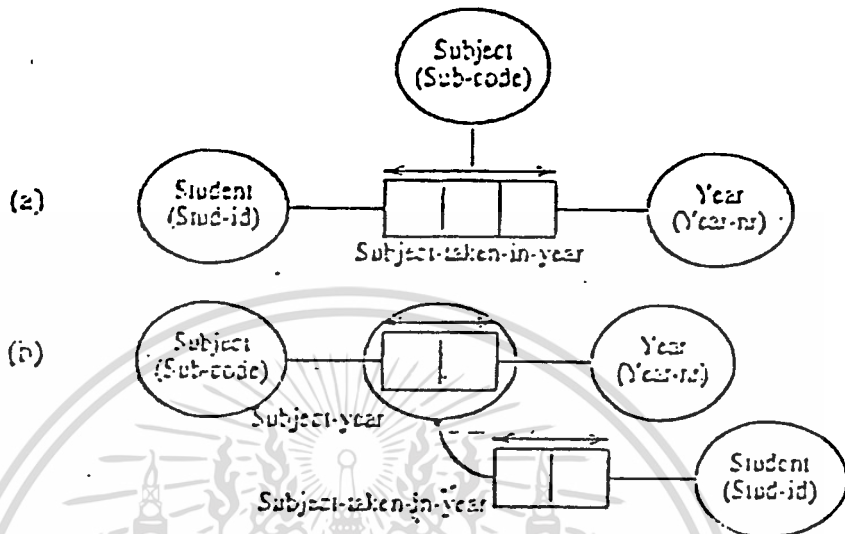
3.2 แต่ละแฟลคส์ที่มีโรลจำนวน n (n -ary, $n \geq 2$) การมีข้อกำหนดความเป็นหนึ่งครอบคลุมโรลจำนวน $n - 1$ จะมีความหมายเหมือนกับแฟลคส์แบบไบนารีที่มีร่างแหไปเกี่ยวข้องกับโรลของสิ่งอื่นแบบร่างแหในระดับที่ $n - 2$ ดังตัวอย่างที่แสดงในรูปที่ 1.4



รูปที่ 1.4 ตัวอย่างคุณสมบัติข้อสอง : แบบแผน (a) เหมือนกับแบบแผน (b)

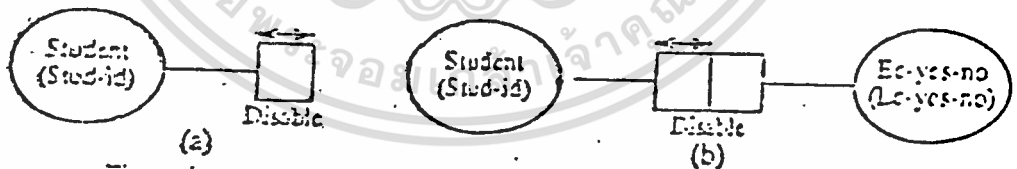
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ 3.3 แต่ละแฟลคส์ที่มีโรลจำนวน n และมีข้อกำหนดความเป็นหนึ่งครอบคลุมทุกโรลของ ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แฟลคส์จะมีความหมายเหมือนกับ แฟลคส์แบบไบนารีที่มีร่างแหไปเกี่ยวข้องกับโรลของสิ่งที่เราสนใจ อันอื่น โดยมีข้อกำหนดแมนเดทอรีปรากฏอยู่บนร่างแห ผลที่ได้จะเป็นร่างแหอันใหม่ออกมา ซึ่งจะ ไปเกี่ยวข้องกับโรลของสิ่งที่เราสนใจอันอื่นต่อๆ ไปอีก ดังแสดงในรูปที่ ข.5.



รูปที่ ข.5. ตัวอย่างคุณสมบัติข้อ 3 : แบบแผน (a) เหมือนกับแบบแผน (b)

3.4 แฟลคส์ที่มีโรลเดียว (Unary) และมีข้อกำหนดความเป็นหนึ่งครอบคลุมอยู่บนโรล นั้นมีความหมายเหมือนกับแฟลคส์ที่มีโรลเดิมไปเกี่ยวข้องกับอีกโรลของสิ่งอื่นที่มีเศษ โดยมีข้อกำหนด แมนเดทอรี และข้อกำหนดความเป็นหนึ่งปรากฏอยู่บนโรลเดิม ดังแสดงในรูปที่ ข.6.



รูปที่ ข.6. ตัวอย่างคุณสมบัติข้อ 4 : แบบแผน (a) เหมือนกับแบบแผน (b)

4. ความหมายของแฟลคส์แบบต่างๆ

ทฤษฎีบทที่ 1

แฟลคส์ที่มีโรลจำนวน n (n - ary, $n \geq 2$) และมีข้อกำหนดความเป็นหนึ่งครอบคลุม ทุกโรลจะแทนมัลติแวลูดีเพนเดนซ์ (Multivalued Dependent : MVD) ของสิ่งที่เราสนใจ

ทฤษฎีบทที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แฟลคส์ที่โวลจำนวน n ($n - \text{ary}, n \geq 2$) และมีข้อกำหนดความเป็นหนึ่งครอบคลุมโวลจำนวน $n - 1$ จะแทนฟังก์ชันนอลดีเพนเดนซ์ (Functional Dependent : FD)

5. แบบแผนรูปแบบปกติที่เหมาะสมที่สุด (Optimal Normal Form : ONF Schema)

เราจะกล่าวว่าแบบแผนระบบฐานข้อมูลแบบสัมพันธ์ (Relational Database Schema) จะเป็นแบบแผนปกติที่เหมาะสมที่สุดก็ต่อเมื่อแบบแผนนั้น ไม่มีการซ้ำซ้อนของข้อมูลและมีจำนวนความสัมพันธ์น้อยที่สุด

6. ขั้นตอนการแปลง (Transformation Algorithm)

ขั้นตอนการแปลงแบบแผนความคิดความเข้าใจรวบยอด ของวิธีวิเคราะห์ข้อมูลแบบอิสระ ให้เป็นแบบแผนความสัมพันธ์แบบเอสคิวแอล (SQL) มี 8 ขั้นตอน คือ

6.1 แปลงยูนิอาร์แฟลคส์ให้เป็น ไบนารีแฟลคส์ที่มีความหมายเหมือนเดิม

6.2 เชื่อมโยงประเภทย่อยของสิ่งที่เราสนใจเข้าหาประเภทที่อยู่สูงกว่า แล้วจัดแมนเดทอรีแฟลคส์ออกจากโวลที่ถูกเชื่อมโยงนี้

6.3 ไม่ต้องสนใจแฟลคส์ที่แสดงถึงตัวที่ใช้ในการอ้างถึงสิ่งที่เราสนใจหรือแสดงชินโนนิมไว้แต่จะมีกล่าวไว้ในขั้นตอนนั้น

6.4 สร้างความสัมพันธ์ขึ้นมา 1 ความสัมพันธ์สำหรับแต่ละแฟลคส์ที่มีข้อกำหนดความเป็นหนึ่งครอบคลุมทุกโวลในแฟลคส์นั้น

6.5 สร้างความสัมพันธ์ขึ้นมา 1 ความสัมพันธ์สำหรับแต่ละแฟลคส์ที่มีจำนวนโวล n โวล ($n > 2$) ซึ่งมีข้อกำหนดความเป็นหนึ่งครอบคลุมโวลจำนวน $n - 1$ โวลในแฟลคส์นั้น

6.6 รวมกลุ่มไบนารีแฟลคส์ที่มีข้อกำหนดความเป็นหนึ่งครอบคลุมเพียงโวลเดียวบนสิ่งที่สนใจร่วมกันอยู่เข้าเป็นความสัมพันธ์จำนวน 1 ความสัมพันธ์ โดยยึดสิ่งที่สนใจร่วมกันนั้นเป็นหลักในการรวมกลุ่ม ถ้าไบนารีแฟลคส์มีข้อกำหนดความเป็นหนึ่งครอบคลุมอยู่บนแต่ละโวล ให้ทำการรวมกลุ่มเข้าหาสิ่งที่สนใจที่มีแมนเดทอรีปรากฏอยู่บนโวลของมัน อย่างไรก็ตาม ถ้าไม่มีแมนเดทอรีปรากฏอยู่บนโวลใด จะทำการรวมกลุ่มเข้าหาสิ่งที่สนใจสิ่งใดก็ได้

6.7 รวมกลุ่มไบนารีแฟลคส์ที่มีสิ่งที่สนใจร่วมกันและมีแมนเดทอรีปรากฏอยู่บนโวลของสิ่งที่สนใจร่วมกันนั้น โดยรวมกลุ่มเข้าหาสิ่งที่สนใจร่วมกันแล้วสร้างความสัมพันธ์ขึ้นมา 1 ความสัมพันธ์

6.8 สร้างความสัมพันธ์ขึ้นมา 1 ความสัมพันธ์สำหรับสิ่งที่สนใจที่มีชินโนนิม แต่ไม่เป็นสิ่งที่สนใจร่วมกันในขั้นตอน 6.6

A-GRAPH (Activity Graph)

หน้าที่ของ A-graph (Activity graph) ใช้ในการอธิบายกิจกรรมในหน่วยงานและระบบสารสนเทศที่เกี่ยวข้อง สัญลักษณ์ที่ใช้ใน A-graph แทนสิ่งต่างๆต่อไปนี้

1 Set ได้แก่ Set ของสิ่งต่างๆ ทั้งบุคคล(Person) วัสดุสิ่งของ(Material) และข่าวสาร(Message) เซ็ตที่เคลื่อนเข้าสู่กิจกรรม เราเรียกว่า อินพุต ส่วนเซ็ตที่เคลื่อนออกจากกิจกรรม เราเรียกว่า เอาต์พุต ของกิจกรรม

2 Activity ทั้งที่กระทำโดยมนุษย์ , เครื่องมืออุปกรณ์ หรือประกอบกัน

3 Flow แสดงการไหลของ Set ระหว่าง Activity ต่างๆ

คำอธิบายสัญลักษณ์ที่ใช้ใน A-graph(ภาพ 3.42 หน้า98 หนังสือ INFORMATION SYSTEM DEVELOPMENT)



REAL SET

: SET ของบุคคล และ สิ่งของ



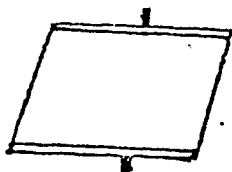
MESSAGE SET

: SET ของ ข่าวสาร เช่นเอกสารต่าง หรือข้อมูลข่าวสารทางโทรศัพท์



COMPOSITE SET

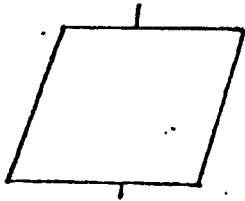
: SET ที่เป็นทั้ง REAL SET และ MESSAGE SET (PERSON/MATERIAL WITH MESSAGE)



REAL FLOW

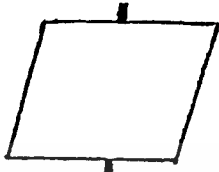
: การไหลของบุคคล หรือวัสดุสิ่งของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



MESSAGE FLOW

: การไหลของข่าวสาร



COMPOSITE FLOW

: การไหลของบุคคล หรือวัตถุสิ่งของ พร้อมข่าวสาร

ACTIVITY

: กิจกรรมในองค์กรนั้นๆ

ตัวอย่างของ A-graph (ใช้รูป 1 หน้า 184 หนังสือคอมพิวเตอร์แมกกาซีน)

A-graph ดังกล่าว เขียนขึ้นจากระบบงานของสหกรณ์ผู้ผลิตนมแห่งหนึ่ง ซึ่งรวมตัวกันขึ้นจากผู้ผลิตหลายราย สหกรณ์นี้มีศูนย์กลางอยู่ที่อหิชาภักตอง อันเป็นสถานที่ประชุมวางแผนการดำเนินงาน คัดสรร ทางผู้ผลิตแต่ละรายจะรับผิดชอบการผลิตและการจำหน่ายเอง การขายปลีกนั้นจะผ่านพ่อค้าคนกลาง ฝ่ายสหกรณ์จะมีการติดสื่อสารกับผู้บริโภคในด้านข่าวสาร เช่น การโฆษณา เป็นต้น ระบบที่เราสนใจ (ระบบที่เกิดปัญหาและต้องแก้ไข) ได้แก่ระบบการรับคำสั่งซื้อและการส่งสินค้าไปยังลูกค้า จากสิ่งต่าง ๆ ดังกล่าว เราเขียนเอ-กราฟได้ดังรูป ซึ่งมีรายละเอียดบางอย่างที่ยังไม่ใส่คำอธิบายถึง ดังนี้คือ

1. บนกระดาษที่เขียนเอ-กราฟจะนิยามกระดาษระบุสิ่งที่จะรู้ โดยแบ่งออกเป็น 3 คอลัมน์อื่นได้แก่

คอลัมน์ที่ 1 ทางซ้ายมือสุด แยกเป็น 3 ส่วนย่อยอีก คือ

- ส่วนที่ 1 บอกชื่อของหน่วยงาน, องค์กร ในที่นี้ สมมติเป็นสมณ.
- ส่วนที่ 2 บอกชื่อบุคคลผู้วิเคราะห์ระบบ (หรือกลุ่ม)
- ส่วนที่ 3 บอกขอบเขตระบบที่เอ-กราฟแสดง

คอลัมน์ที่ 2 ตรงกลาง แยกเป็น 2 ส่วนย่อย

- ส่วนที่ 1 บอกให้ทราบว่าเอกสารนี้ คือ เอ-กราฟ

- ส่วนที่ 2 บอกวันที่ที่เขียนหรือปรับปรุง (ตามเวอร์ชัน)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น กรุณาอย่าเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอลัมน์ที่ 3 ทางขวามือสุด แยกเป็น 2 ส่วนย่อย

- ส่วนที่ 1 นอกเลขอ้างอิงของเอ-กราฟ โดยบ่งถึงกิจกรรมที่เอ-กราฟนั้นอธิบาย
- ส่วนที่ 2 นอกเวอร์ชันที่ปรับปรุง

2. กรอบสี่เหลี่ยมที่ล้อมรอบให้กิจกรรมทั้งหลายอยู่ภายใน ก็คือ ขอบเขตของระบบงานที่เราศึกษา กิจกรรมทั้งหลายที่เราสนใจเป็นกิจกรรมภายในระบบงาน จึงอยู่ภายในกรอบสี่เหลี่ยม แต่สำหรับเซตจะมีอยู่ 3 จำนวน พวกแรกเป็นเซตที่เกิดจากกิจกรรมภายนอกกรอบ แต่เข้ามาเป็นอินพุตของกิจกรรมภายในระบบของเรา เซตพวกนี้จะอยู่ด้านบนนอกกรอบสี่เหลี่ยม ตามตัวอย่าง ได้แก่ เซต 1A และเซต 2A ต่อมาคือพวกที่เป็นเอาต์พุตหรือเกิดจากกิจกรรมภายในระบบ และยังคงใช้งานภายในระบบ (คือ เป็นอินพุตของกิจกรรมอื่นอีกทีหนึ่ง) พวกนี้จะอยู่ภายในกรอบสี่เหลี่ยม ดังเช่น เซต 3A เซต 5B เป็นต้น สำหรับพวกสุดท้ายเป็นเซตที่เกิดจากกิจกรรมในระบบเช่นเดียวกับพวกที่สอง แต่นั่นจากระบบไปเป็นอินพุตของกิจกรรมในระบบอื่นภายนอก เซตพวกนี้จะอยู่ด้านล่างนอกกรอบสี่เหลี่ยม ตามตัวอย่าง ได้แก่ เซต 3C และเซต 6A

3. แต่ละกิจกรรมภายในระบบจะมีหมายเลข เพื่อใช้อ้างอิงที่แตกต่างกันและมีชื่อบ่งบอกไว้ด้วย เช่น กิจกรรมหมายเลข 3 คือ กิจกรรมในออฟฟิศกลาง กิจกรรมหมายเลข 6 คือกิจกรรมของผู้บริหารโลก เป็นต้น สำหรับเซตต่างๆก็มีตัวเลขและตัวอักษรใช้อ้างอิงเช่นเดียวกัน การกำหนดเลขอ้างอิงของกิจกรรมให้กำหนดเลขเรียงลำดับกันจากน้อยไปมาก โดยไล่เรียงกิจกรรมที่มีตำแหน่งบนเอ-กราฟจากบนลงล่าง และจากซ้ายไปขวา ส่วนเซตมีหลักเกณฑ์คือ ดูว่าเซตนั้นเป็นเอาต์พุตของกิจกรรมใด ก็ใช้เลขอ้างอิงของกิจกรรมนั้นประกอบด้วยตัวอักษรภาษาอังกฤษ A, B, ... เป็นตัวอ้างอิงเซต ตัวอย่างเช่น กิจกรรมหมายเลข 3 มีเอาต์พุต 3 เซต ก็จะได้ 3A, 3B และ 3C เป็นตัวอ้างอิงเป็นต้น โดยที่เราไม่ต้องสนใจว่า เซตนั้นอยู่ภายในกรอบสี่เหลี่ยมหรืออยู่นอกกรอบสี่เหลี่ยมด้านล่าง ส่วนเซตที่อยู่ภายนอกกรอบสี่เหลี่ยมด้านบนนั้น ให้ถือเสมือนว่ามันเป็นเอาต์พุตของกิจกรรมอะไรก็ได้ภายนอกกรอบ และกำหนดตัวอ้างอิงขึ้นเอง โดยใช้นหลักเดียวกับที่กล่าวมาแล้ว แต่ห้ามใช้ตัวอ้างอิงที่ซ้ำซ้อน จากตัวอย่างในรูปเช่น เซต 2A ถือเสมือนว่ามันเป็นเอาต์พุตของกิจกรรมหมายเลข 2 ภายนอกกรอบ เป็นต้น

4. รั้วขอบซ้ายของกรอบสี่เหลี่ยมจะนำเลขอ้างอิงของเอ-กราฟ (ซึ่งเขียนอยู่ภายในกรอบสี่เหลี่ยมเล็กกว่าหน้ากระดาษ) มาเขียนกำกับไว้ ตามตัวอย่างนั้นเลขอ้างอิงของเอ-กราฟคือ C0 ก็เขียนตัว C กำกับ (ถ้าเป็นเลข 0 มักจะละไว้ ไม่ลงเป็น C0)

คำอธิบายระบบงานโดยอาศัย A-graph ประกอบ

จากเอ-กราฟในรูป ออฟฟิศกลาง (กิจกรรม 3) จะรับรู้เอาข่าวสารเกี่ยวกับตลาดนมจากภายนอกกรอบ (เซต 1A) ข่าวสารนี้อาจได้จากสื่อมวลชน, ผู้ผลิตนมรายอื่นหรือจากแหล่งข่าวอื่น และยังได้รับรายงานทางด้านการเงินจากสมาชิกของสหกรณ์ (4A) ทางออฟฟิศกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะใช้ในพดทั้งสองเป็นเครื่องช่วยกำหนดแผนและแนวทางดำเนินงานให้กับสมาชิก (3B) และมีการแจ้งข่าวสารความเคลื่อนไหวของกิจกรรมต่างๆ ไปยังลูกค้า คือ ผู้บริโภคและพ่อค้าคนกลาง (3A) รวมทั้งข่าวสารประชาสัมพันธ์ไปยังภายนอกระบบ (3C) เช่น ให้ข่าวแก่หนังสือพิมพ์ธุรกิจ เป็นต้น ทางด้านผู้ผลิตนม (4) จะได้รับวัตถุดิบในการผลิต เช่น น้ำนมดิบ, ส่วนผสม (เช่น น้ำตาล) และบรรจุภัณฑ์ เป็นต้น (2A) ตลอดจนรับแผนงานและแนวทางดำเนินงานจากออฟฟิศกลาง (3B) นอกจากนี้ ยังมี คำสั่งซื้อ, เงินค่าสินค้า และสินค้าส่งกลับคืนจากพ่อค้าด้วย (5B) และแน่นอนครับ สิ่งที่ได้ออกมาจากกิจกรรมนี้ก็คือ สินค้านมที่ผลิตแล้ว ส่งไปยังพ่อค้าพร้อมกับใบอินวอยซ์ (4B) และส่งรายงานด้านการเงินกลับไปยังออฟฟิศกลาง (4A) ไล่ลงมายังพ่อค้าคนกลาง (5) ได้รับสินค้ากันในอินวอยซ์จากผู้ผลิตนม (4B) และรับข่าวสารจากออฟฟิศกลาง เช่น การจัดแคมเปญส่งเสริมการขาย เป็นต้น (3A) แล้วทางพ่อค้าจะส่งนมไปยังผู้บริโภค (5A) และส่งคำสั่งซื้อ, เงินชำระค่าสินค้า และสินค้ากลับคืนไปยังผู้ผลิต (5B) สุดท้าย ได้แก่ ผู้บริโภค (6) ได้รับข่าวสาร เช่น โฆษณาสินค้าโคเร็กซ์ที่แม่ส์จากออฟฟิศกลาง(3A) โดยที่จะมีนมที่ไปซื้อมาไว้บริโภค (6A)

TEXT PAGE

ในการใช้งานจริงแล้ว เอ-กราฟยังมีอุปสรรคและข้อจำกัดอยู่ เพราะหน้ากระดาษมีเนื้อที่จำกัด ผู้เขียนเอ-กราฟจะถูกบังคับทางอ้อม 2 ประการ ได้แก่ ประการแรก การเขียนข้อความใดก็ตามให้สั้นและกะทัดรัดเข้าไว้ อย่างเช่นกิจกรรมหมายเลข 5 ต้องใช้คำว่า "ผู้ผลิตนม" แทนข้อความ "การผลิตและจำหน่ายสินค้าของผู้ผลิตนม" เป็นต้น ส่วนประการที่สองก็คือ หากสิ่งที่เราสนใจ (เซ็ค) นี้นั้น จะต้องพยายามรวมสิ่งเหล่านั้นเข้าไว้เป็นเซ็คเดียวกัน เพื่อให้มีจำนวนเซ็คน้อยที่สุด เช่น เซ็ค 5B แทนที่จะแยกเป็นเซ็คของคำสั่งซื้อ เซ็คของเงินค่าสินค้า และเซ็คของสินค้าส่งคืน ก็ต้องนำมารวมเป็นเซ็คเดียว ข้อจำกัดดังกล่าว ส่งผลให้เอ-กราฟอาจรวบรัดเกินไป จนผู้ทักความเข้าใจได้ยาก กลุ่มไอแซกจึงได้ตกลงกำหนดให้มีการทำเอกสารเสริมขึ้นมาชุดหนึ่ง เรียกว่า Text Page ใช้สำหรับแจกแจงรายละเอียดของเซ็คและของกิจกรรมในเอ-กราฟว่า มีเซ็คและกิจกรรมอะไรบ้าง โดยมีข้อความอธิบายเพิ่มเติมจากข้อความที่เขียนไว้สั้นๆ ในเอ-กราฟ และถ้าเซ็คใดมีส่วนประกอบที่ต้องการแจกแจง ก็อาจเขียนแยกเป็นเซ็คย่อย (subset) ไว้ให้ ตัวอย่าง Text Page ได้แสดงไว้ดังรูป

สผน.

TEXT PAGE

ผู้วิเคราะห์ :

A - GRAPH

วิทยา ต่อศรีเจริญ

เลขอ้างอิง C๘

ขอบเขต :

วันที่ : 13 พย. 2532

ระบบรับคำสั่งซื้อและส่งสินค้า

เวอร์ชัน 4

1A ข่าวสารเกี่ยวกับตลาดนม

1A1 ข่าวสารทางสื่อมวลชน

1A2 ข่าวสารจากแหล่งอื่น

2A วัตถุประสงค์

2A1 มาตรฐานฉบับ

2A2 ส่วนผสม เช่น น้ำตาล เป็นต้น

2A3 บรรทัด

3 การวางแผนและแนวทางการดำเนินงาน ออเนกกลาง

3A ข่าวสารที่ส่งให้กับพ่อค้าคนกลางและผู้บริโภค

3A1 ระยะเวลาสำหรับพ่อค้าคนกลาง (ไปยังกิจกรรมหมายเลข 5)

3A2 ข่าวเรื่องการจัดแคมเปญส่งเสริมการขาย (ไปยังกิจกรรมหมายเลข 5)

3A3 ระยะเวลาสำหรับผู้บริโภค เช่น โดเร่คต์เมลล์ (ไปยังกิจกรรมหมายเลข 6)

3B แผนงานและแนวทางการดำเนินงาน

3B1 แผนงาน

3B2 แนวทางการดำเนินงาน

4 การผลิตและจำหน่ายนมของผู้ผลิตนม

4A รายงานทางการเงินของผู้ผลิตนม

4B สินค้าที่ผลิตแล้วและใบอินวอยซ์ที่จะส่งไปให้พ่อค้าคนกลาง

4B1 เมลล์สำเร็จรูป

4B2 ใบอินวอยซ์

- 5 การรับและเก็บสินค้า ณ ที่ประกอบการของพ่อค้าคนกลาง
- 5A สินค้า (นม) ที่เก็บเอาไว้ในห้องเย็น รอให้ผู้บริโภคซื้อไป
- 5B กำลั้งซื้อ, เงินค่าสินค้า และสินค้าที่ถูกส่งกลับคืน
- 5B1 กำลั้งซื้อ อาจเป็นใบสั่งซื้อหรือสั่งทางโทรศัทพ์
- 5B2 เงินชำระค่าสินค้า หลังจากทีพ่อค้าคนกลางได้รับสินค้าแล้ว
- 5B3 สินค้าที่เจือปนไขไม่ตรงตามทีส่งถูกส่งคืน

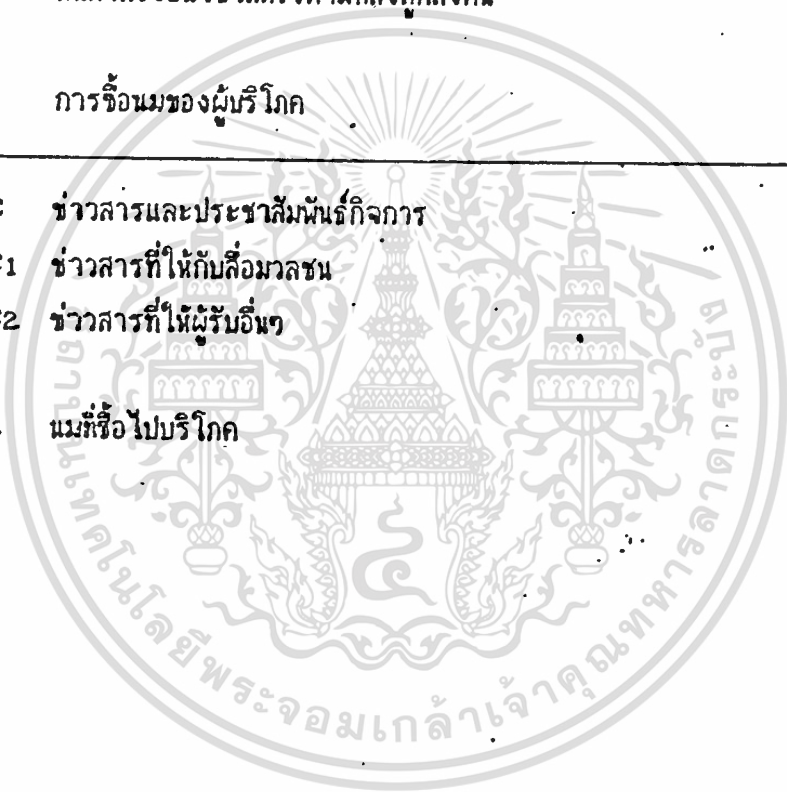
6 การซื้อนมของผู้บริโภค

3C ช่าวสารและประชาสัมพันธ์กิจการ

3C1 ช่าวสารทีให้กับสื่อมวลชน

3C2 ช่าวสารทีให้ผู้รับอื่นว

6A นมทีซื้อไปบริโภค



ข้อกำหนดในการเขียน Text Page มีเป็นข้อๆ ดังนี้

1. เท็กซ์เพจมีหัวกระดาษเช่นเดียวกับเอ-กราฟ แต่เพิ่มคำว่าเท็กซ์เพจตรงกลางหัวกระดาษ
2. ตัวเนื้อหาของเท็กซ์เพจแบ่งเป็น 3 ส่วน แต่ละส่วนแยกให้เห็นชัดจากกันด้วยเส้นตรงแนวนอน โดยที่
 - ส่วนที่ 1 อธิบายรายละเอียดของสิ่งที่อยู่นอกกรอบสี่เหลี่ยมด้านบนของเอ-กราฟ ในที่นี้ ได้แก่ เซ็ต 1A และเซ็ต 2A
 - ส่วนที่ 2 อธิบายรายละเอียดของสิ่งที่อยู่ภายในกรอบสี่เหลี่ยมของเอ-กราฟ ในที่นี้ ได้แก่ กิจกรรมหมายเลข 3, 4, 5 และ 6 ตลอดจนเซ็ต 3A, 3B, 4A, 4B, 5A และ 5B
 - ส่วนที่ 3 อธิบายรายละเอียดของสิ่งที่อยู่ภายนอกกรอบสี่เหลี่ยมด้านล่างของเอ-กราฟ ในที่นี้ ได้แก่ เซ็ต 3C และเซ็ต 6A
3. แต่ละเซ็ตภายในเอ-กราฟสามารถแบ่งเป็นเซ็ตย่อยเพื่อเจาะลึกถึงรายละเอียดของเซ็ตนั้นได้ (โดยที่เซ็ตย่อยเหล่านั้นไม่ได้แสดงไว้ในเอ-กราฟ) ตัวอย่าง ตามรูปที่ 1 เซ็ต 1A แบ่งได้เป็นเซ็ต 1A1 และเซ็ต 1A2 ส่วนเซ็ต 2A ก็แบ่งเป็นเซ็ต 2A1, 2A2 และ 2A3 แต่ไม่แสดงไว้ มาดูจากแจงไว้ที่รูปที่ 2 การแบ่งเซ็ตย่อยจะแบ่งเท่าไรก็ขึ้นอยู่กับผู้เขียน เอ-กราฟเอง บางเซ็ตอย่างเช่นเซ็ต 4A และเซ็ต 5A อาจไม่ต้องแบ่งก็ได้
4. ในการเขียนจะเรียงลำดับหมายเลขอ้างอิงของเซ็ตและของกิจกรรมจากน้อยไปมากภายในส่วนเดียวกัน (หมายถึง ส่วนทั้งสามที่ได้อธิบายไปในข้อ 2) ตามรูปที่ 2 นั้นส่วนที่ 1 เขียน 1A ก่อน 2A ส่วนที่ 3 ก็เขียน 3C ก่อน 6A สำหรับส่วนที่ 2 จะเขียนกิจกรรมที่มีเลขอ้างอิงน้อยก่อน ตามด้วยเซ็ตที่เป็นเอาต์พุตของกิจกรรมนั้น แล้วจึงเขียนกิจกรรมที่มีหมายเลขอ้างอิงถัดไปเรื่อยๆ (โปรดดูตัวอย่างประกอบ)
5. จะสังเกตเห็นว่าเวลาเขียนถึงเซ็ตและกิจกรรมที่มีอ้างอิงไว้ในเอ-กราฟ จะมีการเว้นบรรทัดเสมอ เฉพาะเซ็ตย่อยซึ่งเรามีอ้างอิงไว้ในเอ-กราฟจึงจะเขียนอยู่ใต้เซ็ตใหญ่ของมัน โดยไม่มีการเว้นบรรทัด จะเห็นว่าเท็กซ์เพจมีประโยชน์ต่อเอ-กราฟอย่างมากในแง่ของการเป็นตัวเสริมให้เอ-กราฟมีประสิทธิภาพในการสื่อความหมาย ให้ผู้ดูสามารถเข้าใจระบบได้กระจ่างขึ้น

Overview Graph และ Detail Graph

จุดเด่นประการหนึ่งของ A-graph ก็คือสามารถจำลอง หรืออธิบายการทำงานของระบบได้ละเอียดมากเท่าที่ผู้วิเคราะห์ต้องการ A-Graph ที่ถูกเขียนขึ้นมาแรกสุด และมีความละเอียดน้อยที่สุด ก็คือ Overview Graph จากนั้นถ้าผู้วิเคราะห์ต้องการอธิบายส่วนย่อยต่างๆ ให้ชัดเจนขึ้นอีกก็สามารถจะกระทำได้โดยการเขียน Detail Graph ให้ละเอียดคลึงลงไปเรื่อยๆ เป็นลำดับไป มีข้อสังเกตอยู่ว่า Overview Graph ของแต่ละระบบจะมีเพียงระดับเดียวและมีเพียงภาพเดียว แต่ Detail Graph อาจจะมีได้หลายระดับ และแต่ละระดับก็อาจมีได้หลายภาพ ตัวอย่างของ Detail Graph ได้แสดงไว้ดังรูป (รูป 1, 2 คอมแมทรินเล่ม 12 Page 128, 129)

จากไอเวอร์วิวกภาพของระบบรับคำสั่งซื้อและส่งสินค้า ต้องการขยายรายละเอียดของกิจกรรมหมายเลข 4 "ผู้ผลิตนม" จะเขียนเอ-กราฟและเก็ทซ์เพจได้ดังรูป ซึ่งจะขออธิบายดังนี้ - กรอบสี่เหลี่ยมใหญ่ก็เป็นเช่นเดียวกับไอเวอร์วิวกภาพ คือเป็นการแสดงขอบเขตของระบบที่สนใจ ซึ่งตามรูปนี้ขอบเขตดังกล่าวคือ กิจกรรมหมายเลข 4 "ผู้ผลิตนม" ของไอเวอร์วิวกภาพภายในกรอบแบ่งเป็นกิจกรรมของแผนกต่าง ๆ นั้นหมายความว่ามีการแบ่งแผนกของผู้ผลิตนม ซึ่งที่จริงจะแบ่งกิจกรรมเป็นอย่างไรก็ได้ เช่น อาจรวมหลายแผนกเป็นกิจกรรมเดียวกัน หรือแผนกเดียวแต่แยกเป็นหลายกิจกรรม ทั้งนี้ขึ้นอยู่กับวิจรกิจการของผู้วิเคราะห์ นอกจากนี้ดังได้กล่าวไว้แล้วว่า ดิเรลด์กราฟมิได้หลายระดับ ดังนั้นผู้วิเคราะห์ยังอาจขยายรายละเอียดเพิ่มได้อีก เช่น นำกิจกรรมของแผนกเก็บรักษาและส่งสินค้าไปเขียนเป็นเอ-กราฟภาพใหม่ ซึ่งจะเป็นดิเรลด์กราฟระดับที่ 2 เป็นต้น แต่ในที่นี้เราจะจะไม่เจาะรายละเอียดเหล่านั้นกัน

ข้อจำกัดในการเขียนดิเรลด์กราฟ มีดังต่อไปนี้

1. เลขที่อ้างอิง

เลขที่อ้างอิงของเอ-กราฟจะประกอบด้วยตัวอักษรย่อหน้าหน้า โดยตัวอักษรย่อหน้าจะใช้อ้างอิงถึงเอ-กราฟทุกภาพในระบบ (และจะตั้งขึ้นด้วยความหมายใดก็ได้ เช่น ชื่อย่อกลุ่มผู้วิเคราะห์ระบบ เป็นต้น ในกรณีของ สผน.นี้ คือ ตัว C และมีตัวเลขกำกับท้าย สำหรับไอเวอร์วิวกภาพซึ่งในระบบมีเพียงภาพเดียวจะใช้เลข 0 เสมอ ส่วนดิเรลด์กราฟจะใช้เลขอ้างอิงของกิจกรรมที่นำมาเขียนเป็นดิเรลด์กราฟนั้นเป็นเลขกำกับท้าย จากตัวอย่างเลขที่กำกับท้ายของเลขอ้างอิงคือ เลข 4 เพราะมาจากกิจกรรมหมายเลข 4 (ของไอเวอร์วิวกภาพ)

2. เช็ทซ์ที่อยู่นอกกรอบ

เช็ทซ์ที่อยู่ภายนอกกรอบสี่เหลี่ยมไม่ว่าด้านบนหรือด้านล่างก็ตาม ได้แก่ บรรดาเช็ทซ์หรือเช็ทซ์ย่อยที่เล็ดลอดเข้ามาแล้วจากเอ-กราฟหรือเก็ทซ์เพจที่อยู่เหนือดิเรลด์กราฟนี้ (กรณีนี้เอ-กราฟที่วิเคราะห์ ไอเวอร์วิวกภาพ) จะต้องครอบคลุมเช็ทซ์ด้วย ตามไอเวอร์วิวกภาพ CO เมื่อทราบว่าตัวเลข 2A, 3B และ 5B เป็นอินพุต ส่วนเช็ทซ์ 4A และ 4B เป็นเอาต์พุตของกิจกรรมผู้ผลิตนม ดังนั้นดิเรลด์กราฟ C4 ในรูปที่ 1 จะต้องมีเช็ทซ์ 2A, 3B และ 5B เป็นอินพุตที่ด้านบน นอกกรอบสี่เหลี่ยม ส่วนเช็ทซ์ 4A และ 4B เป็นเอาต์พุตที่ด้านล่างนอกกรอบสี่เหลี่ยม จะมีเช็ทซ์ที่เกินภาพหรือขาดไปจากนี้ไม่ได้ แต่เนื่องจากว่าเช็ทซ์ 3B, 5B และ 4B ได้แยกเป็นเช็ทซ์ย่อยไว้ที่เก็ทซ์เพจของไอเวอร์วิวกภาพ CO อยู่แล้ว จึงสามารถแสดงการแจกแจงแยกเช็ทซ์ย่อยเหล่านั้นไปตามกิจกรรมต่าง ๆ ของดิเรลด์กราฟ C4 ได้

3. เช็ทซ์ที่อยู่นอกกรอบ

เช็ทซ์และกิจกรรมทั้งหลายภายในกรอบสี่เหลี่ยม จะต้องตั้งหมายเลขอ้างอิงของมันด้วยหลักเกณฑ์เดียวกับที่กล่าวไปแล้ว แต่มิได้หมายความว่าแต่ละเอ-กราฟจะมีหมายเลขอ้างอิง 1, 2

เอกสารนี้เป็น หรือ 1A, 1B, 1C เหมือนหรือซ้ำกัน ขอให้สังเกตเห็นเครื่องหมาย "A" หน้าตัวเลขอ้างอิงเหล่านี้

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั้น เครื่องหมายนั้นบอกให้รู้ว่า เชื้อหรือกิจกรรมนั้น ๆ ได้ละหมายเลขอ้างอิงที่อยู่ข้างหน้ามันเอาไว้ หมายเลขนี้ก็คือตัวเลขที่กำกับท้ายเลขอ้างอิงของเอ-กราฟ (ดีเทลด์กราฟ) ซึ่งคือหมายเลขตัวเดียวกับหมายเลขอ้างอิงของกิจกรรมจากเอ-กราฟที่อยู่เหนือเอ-กราฟนี้ซึ่งเรานำมาขยายเป็นเอ-กราฟนี้

จากตัวอย่างหมายเลขที่ละไว้ในตัวอย่างก็คือ เลข 4 ซึ่งมาจาก C4 ของเลขอ้างอิงเอ-กราฟ หรือมาจากกิจกรรมหมายเลข 4 "ผู้ผลิตนม" ของโอเวอร์วิวกราฟ CO ดังนั้นเวลาจะอ้างอิงเชื้อหรือกิจกรรมโดยไม่ให้สับสนแล้ว ก็ต้องอ้างอิงให้ชัดเจน เช่น จะอ้างอิงกิจกรรม "แผนกเก็บรักษาและส่งสินค้า" ก็ต้องบอกว่า กิจกรรมหมายเลข 4 ของเอ-กราฟ C4 หรือเรียกอย่างสั้น ๆ ว่า กิจกรรมหมายเลข 44 (ตัวอย่างในเท็กซ์เพจซึ่งอธิบายความหมายของเชื้อย่อย 3B22)

4. จำนวนกิจกรรม

จำนวนกิจกรรมย่อยในแต่ละเอ-กราฟจะต้องไม่เกินเก้ากิจกรรม มิฉะนั้นถ้าหากมีการอ้างอิงถึงกิจกรรมที่สืบการใช้ตัวเลขอ้างอิงอาจทำให้เกิดความสับสน

5. เส้นการเคลื่อนที่ในกราฟ

ห้ามเส้นแสดงการเคลื่อนที่ (flow) ของเชื้อเชื่อมต่อระหว่างกิจกรรมหนึ่งไปยังกิจกรรมหนึ่งหรือเชื้อหนึ่งไปยังเชื้อหนึ่งโดยตรง ต้องเชื่อมจากกิจกรรมไปยังเชื้อ หรือจากเชื้อไปยังกิจกรรมเท่านั้น

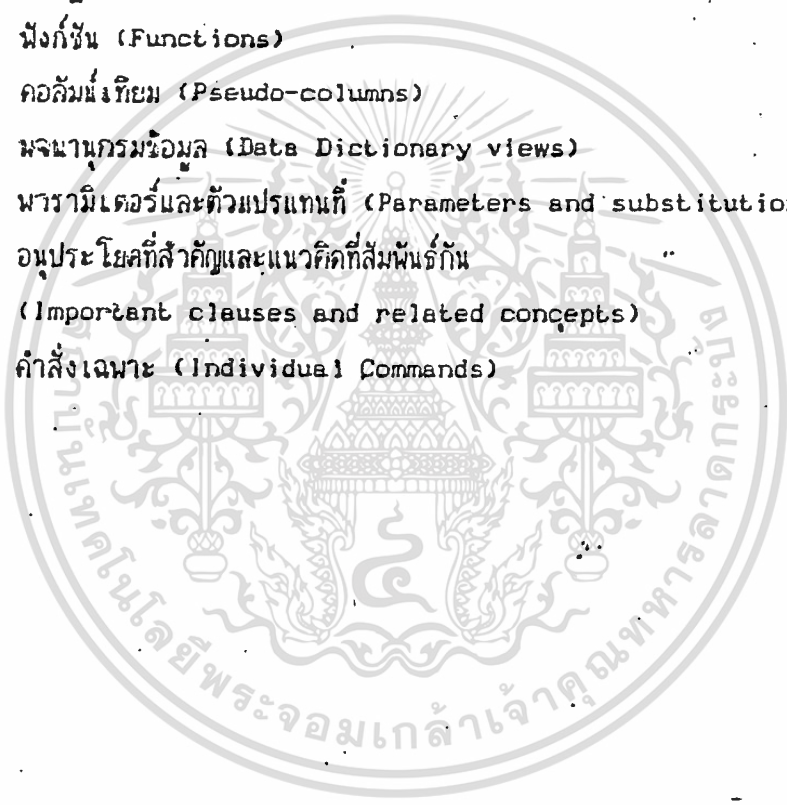
6. ตำแหน่งของ เอ-กราฟ

ในภาควางตำแหน่งของเอ-กราฟและเท็กซ์เพจในสมุดหรือแฟ้มเอกสารนั้น ตามธรรมเนียมที่ยึดถือกันจะจัดให้เท็กซ์เพจอยู่ที่หน้ากระดาษทางซ้ายมือ และเอ-กราฟอยู่หน้าทางขวามือ

เอสคิวแอล*พลัส (SQL*PLUS)

เป็นภาษาที่พัฒนามาจากภาษาเอสคิวแอล เพื่อให้สามารถใช้งานได้กว้างขวางยิ่งขึ้น เช่น การเพิ่มตัวปฏิบัติการให้มากขึ้น มีคำสั่งเฉพาะมากขึ้น เป็นต้น
ลักษณะของภาษาเอสคิวแอลและเอสคิวแอล*พลัสมีดังนี้

1. กฎเกณฑ์การตั้งชื่อ
2. ชื่อไฟล์
3. คำสงวน (Reserved Words)
4. แบบของข้อมูล (Data Types)
5. รูปแบบการแสดงผล (Format Models)
6. ตัวปฏิบัติการ (Operators)
7. ฟังก์ชัน (Functions)
8. คอลัมน์เทียม (Pseudo-columns)
9. พจนานุกรมข้อมูล (Data Dictionary views)
10. พารามิเตอร์และตัวแปรแทนที่ (Parameters and substitution variables)
11. อนุประโยคที่สำคัญและแนวคิดที่สัมพันธ์กัน
(Important clauses and related concepts)
12. คำสั่งเฉพาะ (Individual Commands)



1. กฎเกณฑ์การตั้งชื่อ

ชื่อคือคำที่ตั้งขึ้นเพื่อใช้เป็นชื่อของ เทเบิล (Table), วิว (View), ซินโนนิม (Synonym), คอลัมน์ (Column), อินเด็กซ์ (Index) และตัวแปร ซึ่งมีลักษณะดังนี้

- 1) ชื่อต้องขึ้นต้นด้วยตัวอักษร
- 2) อักขระที่นำมาใช้เป็นชื่อได้คือ A-Z, a-z, 0-9 และ _
- 3) ความยาวของชื่อ มีค่าตั้งแต่ 1 อักขระถึง 30 อักขระ
- 4) อักขรตัวใหญ่และตัวเล็กมีความหมายเหมือนกัน เช่น FRIEND กับ friend เป็นชื่อเดียวกัน
- 5) ชื่อต้องไม่มีอัญประกาศ
- 6) ไม่ซ้ำกับคำสั่งงาน

2. ชื่อไฟล์

คิวรี (Query) และเท็กซ์ (Text) หรือสพูล (Spool) เป็นไฟล์ระบบจัดการ แต่ละไฟล์ประกอบด้วยชื่อและชนิดของไฟล์ ซึ่งชนิดของไฟล์จะถูกกำหนดโดยจุดและส่วนขยาย เช่น

ชนิดของไฟล์	ส่วนขยาย	ตัวอย่าง
คิวรี	.SQL	EMPLOYEE.SQL
ลิสต์ (List) หรือสพูล	ขึ้นกับระบบ	REPORT.LST

3. คำสงวน

ได้

คำต่อไปนี้ เป็นคำที่ถูกลงนามไว้ในเฮสคิวแอล*พลัส (SQL*PLUS) จะนำมาใช้เป็นชื่อไม่

ACCESS	DECIMAL	INCREMENT	NUMBER	SHARE
ADD	DEFAULT	INDEX	OF	SIZE
ALL	DEFINITION	INDEXED	OFFLINE	SMALLINT
ALTER	DELETE	INDEXPAGES	OLD	SPACE
AND	DESC	INITIAL	ON	START
ANY	DISTINCT	INSERT	ONLINE	SUCCESSFUL
APPEND	DOES	INTEGER	OPTIMIZE	SYNONYM
AS	DROP	INTERSECT	OPTION	SYSDATE
ASC	EACH	INTO	OR	SYSSORT
ASSERT	ELSE	IS	OEDER	TABLE
ASSIGN	ERASE	LEVEL	PARTITION	TEMPORARY
AUDIT	EVALUATE	LIKE	PCTFREE	THEN
BETWEEN	EXCLUSIVE	LIST	PRIOR	TO
BY	EXISTS	LOCK	PRIVILEGES	TRIGGER
CHAR	FILE	LONG	PUBLIC	UID
CLUSTER	FLOAT	MAXEXTENTS	RAW	UNION
COLUMN	FOR	MINUS	RENAME	UNIQUE
COMMENT	FORMAT	MODE	REPLACE	UPDATE
COMPRESS	FROM	MODIFY	RESOURCE	USER
CONNECT	GRANT	MOVE	REVOKE	USING
CONTAIN	GRAPHIC	NEW	ROW	VALIDATE
CONTAINS	GROUP	NOAUDIT	ROWID	VALUES
CRASH	HAVING	NOCOMPRESS	ROWNUM	VARCHAR
CREATE	IDENTIFIED	NOLIST	ROWS	VARGRAPHIC
CURRENT	IF	NOSYSSORT	RUN	VIEW
DATAPAGES	IMAGE	NOT	SELECT	WHENEVER
DATE	IMMEDIATE	NOWAIT	SESSION	WHERE
DBA	IN	NULL	SET	WITH

4. แบบของข้อมูล

4.1 การระบุแบบของข้อมูล

แบบของข้อมูล	วิธีการระบุเมื่อสร้างหรือแก้ไขเทเบิล	หมายเหตุ
Char	CHAR(size) VARCHAR(size)	เป็นข้อมูลแบบอักขระ โดยที่ size หมายถึง ความยาวของอักขระ มีค่ามากที่สุดคือ 240 เหมือนกับ CHAR
Date	DATE	วันเดือนปีที่ถูกต้องมีช่วงตั้งแต่วันที่ 1 เดือนมกราคมก่อนคริสต์ศักราช จนถึงวันที่ 31 ธันวาคม ค.ศ.4712
Long	LONG LONG VARCHAR	เป็นข้อมูลแบบอักขระที่มีความยาวถึง 65536 อักขระ กำหนดคอลัมน์แบบ LONG ได้เพียง 1 คอลัมน์เท่านั้นต่อ 1 เทเบิลและไม่ใช่คอลัมน์แบบ LONG ในคิวรีย่อย, ฟังก์ชันนิพจน์, อนุประโยค WHERE หรืออินเด็กซ์ เหมือนกับ LONG
Number	NUMBER NUMBER(size)	คอลัมน์ของตัวเลขกับสเปซไม่เกิน 40 ตัว ไม่นับจุดทศนิยมและเครื่องหมายบวกลบ โดยที่ตัวเลขถูกแสดงได้ 2 แบบคือ 1) เลข '0'-'9' , เครื่องหมาย '+' และ '-' และจุดทศนิยม 2) เลขแบบวิทยาศาสตร์ เช่น 1.85E3 หมายถึง 1850 size หมายถึง ขนาดของคอลัมน์ มีค่าไม่เกิน 105 ตัว

แบบของข้อมูล	วิธีการเมื่อสร้างหรือแก้ไขเทเบิล	หมายเหตุ
	NUMBER(size,dec)	size คือ ขนาดของคอลัมน์ทั้งหมด dec คือ ขนาดของคอลัมน์หลังจุดทศนิยมซึ่งมีขนาดไม่เกิน 240 ตัว
	NUMBER(*)	เหมือนกับ NUMBER
	DECIMAL	เหมือนกับ NUMBER ไม่รับค่า size หรือ dec
	FLOAT	เหมือนกับ NUMBER
	INTEGER	เหมือนกับ NUMBER ไม่รับค่า size หรือ dec
	SMALLINT	เหมือนกับ INTEGER
Raw	RAW(size)	เป็นข้อมูลแบบเลขฐานสอง ซึ่งขนาดความยาวนับเป็นไบนารี (Byte) และมีขนาดไม่เกิน 240 ไบนารี
Long Raw	LONG RAW	เป็นข้อมูลแบบเลขฐานสอง หรือแบบ LONG แบบใดแบบหนึ่ง
RowID	(none)	ค่าที่ใช้ในการพิสูจน์ว่าใครไว้ในเทเบิลไม่ซ้ำกัน

4.2 การเปรียบเทียบข้อมูล

1) ตัวเลข

เลขที่มีค่าในคู่กว่ามีค่ามากกว่า เลขที่มีค่าเล็กกว่าและ เลขลบมีค่าน้อยกว่า เลข

บวก

2) วันเดือนปี

วันเดือนปีล่าสุดมีค่ามากกว่าวันเดือนปีที่อยู่ก่อนหน้านั้น

3) อักขระ

การเปรียบเทียบอักขระเป็นไปตามตารางที่อยู่ถัดไป จากตาราง ตัวเลขหน้าอักขระ คือรหัสของอักขระนั้น ซึ่งรหัสที่ใหญ่กว่ามีค่ามากกว่ารหัสที่เล็กกว่า ในการเปรียบเทียบข้อมูลแบบนี้ อักขระตัวเล็กกับตัวใหญ่มีความหมายต่างกัน สำหรับรหัสแอสกี (ASCII CODE) อักขระตัวเล็ก (a-z) มีค่ามากกว่าอักขระตัวใหญ่ (A-Z) ส่วนรหัสเอ็บซิดิค (EBCDIC CODE) อักขระตัวใหญ่

แบบของข้อมูล	วิธีระบุเมื่อสร้างหรือแก้ไขเทเบิล	หมายเหตุ
	NUMBER(size,dec)	size คือ ขนาดของคอลัมน์ทั้งหมด dec คือ ขนาดของคอลัมน์หลังจุดทศนิยมซึ่งมีขนาดไม่เกิน 240 ตัว
	NUMBER(*)	เหมือนกับ NUMBER
	DECIMAL	เหมือนกับ NUMBER ไม่รับค่า size หรือ dec
	FLOAT	เหมือนกับ NUMBER
	INTEGER	เหมือนกับ NUMBER ไม่รับค่า size หรือ dec
	SMALLINT	เหมือนกับ INTEGER
Raw	RAW(size)	เป็นข้อมูลแบบเลขฐานสอง ซึ่งขนาดความยาวนับเป็นไบนารี (Byte) และมีขนาดไม่เกิน 240 ไบนารี
Long Raw	LONG RAW	เป็นข้อมูลแบบเลขฐานสอง หรือแบบ LONG แบบใดแบบหนึ่ง
RowID	(none)	ค่าที่ใช้ในการพิสูจน์ว่าไว้ในเทเบิลไม่ซ้ำกัน

4.2 การเปรียบเทียบข้อมูล

1) ตัวเลข

เลขที่มีค่าใหญ่กว่ามีค่ามากกว่าเลขที่มีค่าเล็กกว่าและเลขลบมีค่าน้อยกว่าเลขบวก

2) วันเดือนปี

วันเดือนปีล่าสุดมีค่ามากกว่าวันเดือนปีที่อยู่ก่อนหน้ามัน

3) อักขระ

การเปรียบเทียบอักขระเป็นไปตามตารางที่อยู่ถัดไป จากตาราง ตัวเลขหน้าอักขระ คือรหัสของอักขระนั้น ซึ่งรหัสที่ใหญ่กว่ามีค่ามากกว่ารหัสที่เล็กกว่า ในการเปรียบเทียบข้อมูลแบบนี้อักขระตัวเล็กกับตัวใหญ่มีความหมายต่างกัน สำหรับรหัสแอสกี (ASCII CODE) อักขระตัวเล็ก (a-z) มีค่ามากกว่าอักขระตัวใหญ่ (A-Z) ส่วนรหัสเอ็บบีซีดีค (EBCDIC CODE) อักขระตัวใหญ่

มีค่ามากกว่าอักษรตัวเล็ก การใช้ตารางหนึ่งตารางได้ขึ้นกับว่าเครื่องคอมพิวเตอร์ที่ใช้ นั้น ใช้รหัสแบบใด

ตาราง 4-1 แสดงรหัสแอสกี (ASCII)

1. blank	28. ;
2. !	29. <
3. "	30. =
4. #	31. >
5. \$	32. ?
6. %	33. @
7. &	34-59. A-Z
8. ' .	60. [
9. (61. \
10.)	62.]
11. *	63. ^
12. +	64. _
13. ,	65. `
14. -	66-91. a-z
15. .	92. {
16. /	93.
17-26. 0-9	94. }
27. :	95. ~

ตาราง 4-2 แสดงรหัสเอ็บซีดีค (EBCDIC)

1. blank	16. /
2. 0	17. ,
3. .	18. X
4. <	19. _
5. (20. >
6. +	21. ?
7.	22. :
8. &	23. #
9. !	24. @
10. \$	25. '
11. *	26. =
12. >	27. "
13. {	28-53. a-z
14. [54-79. A-Z
15. -	80-89. 0-9

5. รูปแบบการแสดงผล

5.1 รูปแบบการแสดงผลของอักขระ

ประกอบด้วยตัวอักษร 'A' ตามด้วยความกว้างของคอลัมน์ที่เป็นตัวอักษร ถ้าค่าที่จะแสดงเกินความยาวของคอลัมน์ ไม่พอดีกัน จะมีการตัดอักขระที่เกินทิ้งหรือขยายความกว้างที่จะแสดงผลเพื่อรองรับอักขระที่เกินออกมาเพื่อจะได้แสดงผลได้หมด ขึ้นกับว่า SET WRAP OFF หรือ SET TRUNCATE ON สำหรับการรับอักขระทั้ง ส่วนการขยายความยาวของคอลัมน์ ต้องใช้คำสั่ง SET WRAP ON หรือ SET TRUNCATE OFF

5.2 รูปแบบการแสดงผลของตัวเลข

อีเลเมนส์	ตัวอย่าง	คำอธิบาย
9	9999	จำนวนตัวเลขกำหนดความกว้างที่จะแสดงผล
0	0999	แสดงเลขศูนย์นำหน้า
*	*9999	เติมเครื่องหมายดอลลาร์ ('\$') หน้าตัวเลข
B	B9999	แสดงค่า 0 เป็นแบลิ่งค์ (Blank) ไม่ใช่ '0'
MI	9999MI	แสดงค่า '-' หลังค่าที่เป็นลบ
PR	9999PR	แสดงค่าที่เป็นลบไว้ภายในเครื่องหมาย < และ >

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในองค์กรที่มอบหมายไปและจะไม่อนุญาตให้เผยแพร่ภายนอกได้ภายใต้เงื่อนไขการดำเนินการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

99.99 แสดงจุดทศนิยมตามตำแหน่งที่ระบุ
 V 999V99 คุณค่าตัวเลขด้วย 10ⁿ โดยที่ n คือตัวเลขที่อยู่หลัง V
 E 9.999EEEE แสดงตัวเลขแบบวิทยาศาสตร์ รูปแบบนี้ต้องมี E สี่
 ตัว

วันเดือนปี DATE แสดงวันเดือนปีในรูปแบบ 'MM/DD/YY'

5.3 รูปแบบการแสดงผลของวันเดือนปี

รูปแบบ	ความหมาย
SCC หรือ CC	ศตวรรษ ถ้ามี 'S' อยู่ข้างหน้าวันเดือนปีก่อนคริสต์ศักราช
YYYY หรือ SYYYY	ปี ถ้ามี 'S' อยู่ข้างหน้าปีก่อนคริสต์ศักราช
YYY YY หรือ Y	แสดงตัวเลข 3,2 หรือ 1 ตัวหลัง ของค่าปี
Y,YYY	แสดงปีและแสดงลูกน้ำในตำแหน่งที่ระบุ เช่น 1,990
SYEAR หรือ YEAR	ปี ถ้ามี 'S' อยู่ข้างหน้าปีก่อนคริสต์ศักราช
BC หรือ AD	เป็นตัวชี้ว่าเป็น BC หรือ AD
B.C. หรือ A.D.	เป็นตัวชี้ว่าเป็น B.C. หรือ A.D.
Q	หนึ่งในสี่ของปี คือ สามเดือน
MM	เดือน
MONTH	ชื่อของเดือน ความยาว 9 อักขระ (นับสเปซด้วย)
MON	ชื่อของเดือน ย่อเหลือ 3 อักขระ
WW หรือ W	สัปดาห์ของปีหรือเดือน
DDD DD หรือ D	วันของปี เดือน หรือสัปดาห์
DAY	ชื่อของวัน ความยาว 9 อักขระ (นับสเปซด้วย)
DY	ชื่อของวัน ย่อเหลือ 3 อักขระ
J	วันแบบของยูเลียน คือนับจำนวนวันตั้งแต่ 31 ธันวาคม ก่อนคริสต์ศักราช
AM หรือ PM	เป็นตัวชี้ว่าเป็น AM หรือ PM
A.M. หรือ P.M.	เป็นตัวชี้ว่าเป็น A.M. หรือ P.M.
HH หรือ HH12	นับชั่วโมงแบบ 1-12
HH24	นับชั่วโมงแบบ 0-23
MI	นาที
SS	วินาที
SSSSS	วินาที หลังจากเที่ยงคืน

เอกสารนี้เป็นเอกสารที่สงวนไว้/ etc. ใช้งานเพื่อการเว้นวรรคเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"..." สตรีง

อุปสรรค (Prefix) ที่จะถูกเติมเข้ากับรูปแบบข้างบนได้แก่ fm ซึ่งย่อมาจากคำว่า "Fill Mode" สำหรับเติมหน้า MONTH หรือ DAY เท่านั้น

อาคม (Suffix) ที่จะถูกเติมเข้ากับรูปแบบข้างบนได้แก่

TH หมายถึง เลขลำดับ เช่น รูปแบบ DDTH แสดงผลเป็น 4TH

SP หมายถึง เลขที่สะกดตามปกติ เช่น รูปแบบ DDSP แสดงผลเป็น FOUR

SPTH หรือ THSP หมายถึง เลขที่สะกดแบบลำดับ เช่น รูปแบบ DDSPTH

แสดงผลเป็น FOURTH

รูปแบบการแสดงผลของอักขระ และตัวเลขใช้กับคำสั่ง BTITLE, TTITLE, COLUMN เพื่อควบคุมการแสดงผลของข้อมูล ส่วนรูปแบบการแสดงผลของวันเดือนปี ใช้กับฟังก์ชัน TO_CHAR และ TO_DATE

6. ตัวอย่างปฏิบัติการ

6.1 ตัวอย่างปฏิบัติการที่แสดงความสัมพันธ์ในประโยคของเอสคิวแอล*ผลลัพธ์

ตัวอย่างปฏิบัติการ	คำอธิบาย	ตัวอย่าง
-	ยังไม่จบคิวรี คือ	COLUMN JOB -
-	ยังมีคำสั่งอื่นต่ออีก	HEADING 'Job Title';
&	กำหนดพารามิเตอร์ในไฟล์	SELECT * FROM EMP
&	คำสั่ง ซึ่งจะรันโดยคำสั่ง	WHERE JOB='&2' AND
&	START	SAL=&1;
&, &&	กำหนดตัวแปรแทนที่ในคำสั่งของเอสคิวแอล	SELECT * FROM EMP
&	ค่าปัจจุบันของตัวแปรถูกแทนด้วยชื่อของตัวแปรในแต่ละครั้ง	WHERE JOB='&WHICH_JOB' AND SAL=&SALARY;
()	กร่อมคิวรีย่อย	SELECT *_FROM EMP WHERE COMP> (SELECT COMP FROM EMP WHERE ENAME='JONES');
"	กร่อมอักขระหรือวันเดือนปี	'Brien' '17-JAN-86'
"	กร่อมชื่อคอลัมน์ที่มีอักขระพิเศษ	SELECT NAME "EMP NAME", EMPNO "EMP NUM", FROM EMP;

NOT IN	ไม่ความหมายเหมือนกับ 'I=ALL'	...WHERE SAL NOT IN (SELECT SAL FROM EMP WHERE DEPTNO=30)
ANY	เปรียบเทียบค่ากับแต่ละ ค่าที่ถูกส่งมาจากลิสต์หรือ คิวรีย่อย	...WHERE SAL = ANY (SELECT SAL FROM EMP WHERE DEPTNO=30)
ALL	เปรียบเทียบค่ากับทุกๆ ค่าที่ถูกส่งมาจากลิสต์หรือ คิวรีย่อย	...WHERE SAL > ALL (SELECT SAL FROM EMP WHERE DEPTNO=30)
[NOT] BETWEEN ...AND...	[ไม่]มากกว่าหรือเท่ากับ ค่าแรกและน้อยกว่าหรือ เท่ากับค่าหลัง	...WHERE A BETWEEN 1 AND 9
EXISTS	เป็นจริงถ้าคิวรีย่อยส่ง กลับมาอย่างน้อย 1 โรว์	...WHERE EXISTS (SELECT SAL FROM EMP WHERE DEPTNO=30)
[NOT] LIKE	[ไม่]เหมือนกับค่าที่มีรูปแบบ หลัง LIKE	...WHERE CODE LIKE 'TX'
'X'	คือสตริงของอักขระใดๆ	
'-'	คืออักขระใดๆ 1 ตัว	
IS [NOT] NULL	เป็น [ไม่]เป็น NULL	...WHERE JOB IS NULL
NOT	แปลงผลที่ได้ทางตรรก จากการตรวจสอบ นิพจน์ ให้เป็นค่า ตรงกันข้าม	...WHERE NOT (JOB IS NULL) ...WHERE NOT (SAL BETWEEN 500 AND 999)
AND	รวมค่าทางตรรกจากการ ตรวจสอบนิพจน์ชายและขวา ของ AND ซึ่งค่าทางตรรกจะ เป็นจริงถ้าค่าทั้งสองเป็นจริง	...WHERE A=1 AND B=1
OR	รวมค่าทางตรรกจากการ ตรวจสอบนิพจน์ชายและขวา ของ OR ซึ่งค่าทางตรรกจะ เป็นจริงถ้ามีค่าหนึ่งเป็นจริง	...WHERE A=1 OR B=1

6.4 ตัวปฏิบัติการกับนิพจน์ของคิวรี

ตัวปฏิบัติการ	คำอธิบาย	ตัวอย่าง
()	คร่อมคู่ที่ทำปฏิบัติการต่อกัน	SELECT... UNION (SELECT... MINUS SELECT...);
UNION	โรว์ที่เป็นของคิวรีทางซ้าย หรือโรว์ที่เป็นของคิวรีทางขวาของ UNION หรือโรว์ของคิวรีทั้งสองโดยที่โรว์ที่ได้มานั้นต้องแตกต่างกัน	...SELECT... UNION SELECT...;
INTERSECT	โรว์ที่เป็นของคิวรีทั้งสอง	...SELECT... INTERSECT SELECT...;
MINUS	โรว์ที่เป็นของคิวรีทางซ้ายของ MINUS และต้องไม่เป็นโรว์ของคิวรีทางขวาของ MINUS	...SELECT... MINUS SELECT...;

6.5 ตัวปฏิบัติการอื่นๆในภาษาเอสคิวแอล

ตัวปฏิบัติการ	คำอธิบาย	ตัวอย่าง
(+)	กำหนดว่าคอลัมน์ที่อยู่ข้างหน้า เป็นคอลัมน์ของเอาท์เทอร์จอย	...WHERE DEPT.DEPTNO = EMP.DEPTNO (+)
*	เลือกทุกคอลัมน์จากเทเบิลในคิวรี	...SELECT *... ...COUNT(*)...
ALL	เก็บค่าที่ซ้ำกันในคิวรีและนำมารวมกัน	...SELECT EMP.*... ...COUNT(ALL DEPTNO)...
DISTINCT	กำจัดโรว์ที่ซ้ำกันจากผลที่ได้ของคิวรีหรือกำจัดค่าที่ซ้ำกันจากนิพจน์ที่นำมารวมกัน	...SELECT ALL *... ...COUNT(DISTINCT DEPTNO)...
PRIOR	กำหนดความสัมพันธ์แบบพ่อ-ลูกในคิวรี นิพจน์ที่อยู่หน้า PRIOR แทนพ่อส่วนนิพจน์ที่อยู่หลัง แทนลูก	...CONNECT BY PRIOR EMPNO=MGR

7. ฟังก์ชัน

จะใช้รูปแบบของข้อมูลของฟังก์ชันอย่างย่อๆ กล่าวก็คือ

- 'C' หมายถึง แบบของข้อมูลแบบ Char
- 'D' หมายถึง แบบของข้อมูลแบบ Date
- 'N' หมายถึง แบบของข้อมูลแบบ Number
- 'R' หมายถึง แบบของข้อมูลแบบ row ID
- 'W' หมายถึง แบบของข้อมูลแบบ Raw
- '*' ขึ้นกับคำอธิบายฟังก์ชัน

n, m หมายถึง นิพจน์ตัวเลข

char, char1, char2, set, from, to หมายถึง นิพจน์อักขระ

d, e หมายถึง นิพจน์วันเดือนปี

expr, search, return, default หมายถึง นิพจน์ของแบบของข้อมูลแบบใดแบบหนึ่ง

ฟังก์ชันในเอสคิวแอลจะผลิตแบ่งเป็น

7.1 ฟังก์ชันที่เกี่ยวข้องกับตัวเลข

แบบของข้อมูล	ฟังก์ชัน	ผลที่ได้จากฟังก์ชัน
N	ABS(n)	ค่าสัมบูรณ์ของ n
N	CEIL(n)	เลขจำนวนเต็มที่มีน้อยที่สุดซึ่งมากกว่าหรือเท่ากับ n
N	FLOOR(n)	เลขจำนวนเต็มที่มีมากที่สุดซึ่งเท่ากับหรือน้อยกว่า n
N	MOD(n)	เศษที่เกิดจากการหาร n ด้วย m
N	POWER(n)	m ยกกำลัง n ถ้า n ไม่ใช่เลขจำนวนเต็ม จะตัดเศษทิ้ง
N	ROUND(n)	เติมตำแหน่งทศนิยมใน n เป็นจำนวน m ตำแหน่ง ถ้าไม่ใส่ค่า m ให้ m เป็น 0 และถ้า m เป็นค่าลบ ให้เติมเลขทางซ้ายของจุดทศนิยม
N	SIGN(n)	ถ้า $n < 0$ ผลที่ได้คือ -1 ถ้า $n = 0$ ผลที่ได้คือ 0 ถ้า $n > 0$ ผลที่ได้คือ 1
N	SQRT(n)	ค่ารากที่สองของ n ถ้า $n < 0$ ผลที่ได้คือนักล
N	TRUNC(n)	ตัดค่าทศนิยมทิ้ง n เป็นจำนวน m

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่ควรนำออกไปใช้โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่ง ถ้าไม่ใช่ค่า m ให้ m เป็น ๑ และ ถ้า m เป็นค่าลบ ให้ค่าเลขทางซ้ายของจุดทศนิยมเป็น ๑ โดยการตัดเลขทางซ้ายของจุดทศนิยมทิ้ง

7.2 ฟังก์ชันที่กระทำกับอักขระ

แบบของข้อมูล	ฟังก์ชัน	ผลที่ได้จากฟังก์ชัน
N	ASCII(char)	ค่ารหัสแอสกีของอักขระแรกของ char
C	CHR(n)	อักขระซึ่งมีค่ารหัสแอสกีตรงกับ n
C	INITCAP(char)	อักขระตัวแรกของอักขระใน char
N	INSTR(char1, char2[, n[, m]])	ตำแหน่งของตัวที่ m ของ char2 ใน char1 และเริ่มค้นหาที่ตำแหน่ง n ถ้าไม่ใช่ค่า m, n ให้ m, n เป็น 1, 1 ตามลำดับ ตำแหน่ง n ที่ให้มาต้องสัมพันธ์กับอักขระแรกของ char1 และ $n > 1$
N	LENGTH(char)	ความยาวของ char
C	LOWER(char)	แปลงอักขระตัวใหญ่ใน char ให้เป็นตัวเล็ก
C	LPAD(char1, n[, char2])	เติมทางซ้ายของ char1 จนถึงความยาว n ด้วยอักขระใน char2 เติมซ้ำกันหลายครั้งเท่าที่จำเป็น ถ้าไม่ใช่ char2 ให้เติมสเปซแทน
C	LTRIM(char, set)	ลบอักขระเริ่มต้นใน char ออกจนกระทั่งอักขระตัวแรกไม่อยู่ใน set
C	RPAD(char1, n[, char2])	เติมทางขวาของ char1 จนถึงความยาว n ด้วยอักขระใน char2 เติมซ้ำกันหลายครั้งเท่าที่จำเป็น ถ้าไม่ใช่ char2 ให้เติมสเปซแทน
C	RTRIM(char, set)	ลบอักขระสุดท้ายใน char ออกจนกระทั่งอักขระตัวหลังสุดไม่อยู่ใน set
C	SOUNDEX(char)	ค่าของอักขระแทนการออกเสียงค่าใน char
C	SUBSTR(char, m[, n])	สตริงย่อยของ char ซึ่งอักขระของสตริงย่อยเริ่มที่ตำแหน่ง m มีความยาว n

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อักขระและถ้าไม่ใช่ n แล้วอักขระสุดท้ายของสตริงย่อยจะเป็นตัวเดียวกับอักขระสุดท้ายของสตริง char

- C TRANSLATE(char, from, to) ย้ายชุดอักขระจาก from ไป to โดยที่แต่ละอักขระใน char ซึ่งปรากฏใน from จะถูกย้ายไปเป็นอักขระที่ตรงกันใน to
- C UPPER(char) แปลงอักษรตัวเล็กใน char ให้เป็นตัวใหญ่
- C USERENV(char) ข้อมูลเกี่ยวกับของผู้ใช้ซึ่งเป็นประโยชน์ในการเขียนเทเบิลที่ใช้ในการตรวจสอบโดยเฉพาะ

7.3 ฟังก์ชันที่ทำงานกับกลุ่ม

แบบของข้อมูล	ฟังก์ชัน	ผลที่ได้จากฟังก์ชัน
N	AVG([DISTINCT ALL]n)	ค่าเฉลี่ยของ n โดยไม่สนใจค่าที่เป็นนัลล์
N	COUNT([DISTINCT ALL]expr[*])	จำนวนโรว์ซึ่งนิพจน์ expr กำหนดบางอย่างซึ่งไม่ใช่ นัลล์และ '*' ทำให้ COUNT นับโรว์ที่เลือกมาทั้งหมด
N	MAX([DISTINCT ALL]expr)	ค่าที่มากที่สุดของนิพจน์
N	MIN([DISTINCT ALL]expr)	ค่าที่น้อยที่สุดของนิพจน์
N	STDDEV(DISTINCT ALL]n)	ส่วนเบี่ยงเบนมาตรฐานของ n โดยไม่สนใจค่าที่เป็นนัลล์
N	SUM([DISTINCT ALL]n)	ผลรวมของค่าของ n
N	VARIANCE([DISTINCT ALL]n)	ความแปรปรวนของ n โดยไม่สนใจค่าที่เป็นนัลล์

7.4 ฟังก์ชันที่เกี่ยวกับการแปลงค่า

แบบของข้อมูล	ฟังก์ชัน	คำอธิบาย
C	CHARTOROWID(char)	แปลงข้อมูลแบบ Char ไปเป็นแบบ row ID
H	HEXTORAW(char)	แปลงค่าของอักขระซึ่งเป็นเลขฐานสิบหกไปเป็นเลขฐานสอง
C	RAWTOHEX(row)	แปลงข้อมูลแบบ Row ไปเป็นค่าของอักขระ

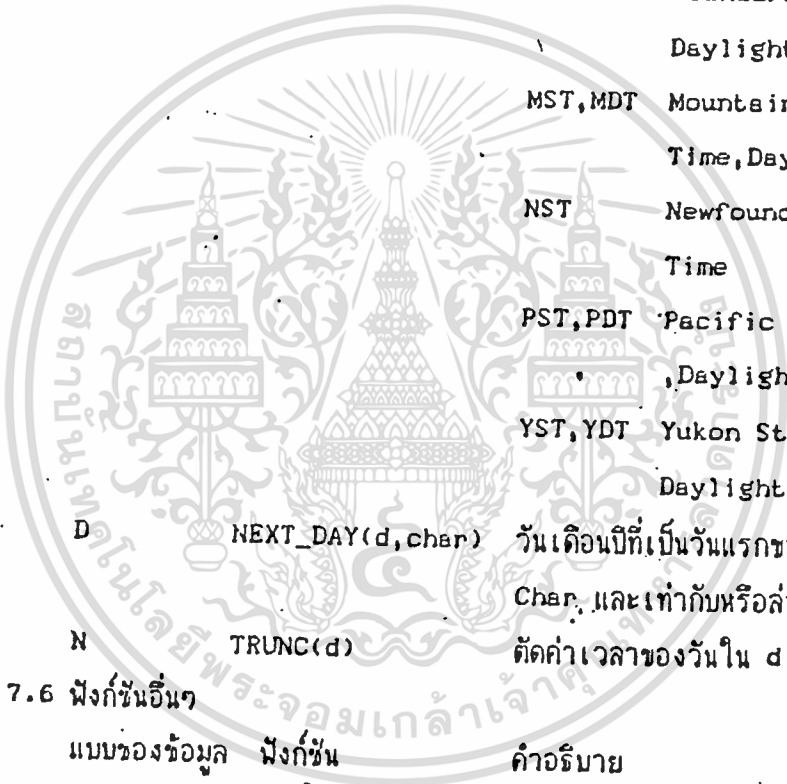
ชระซึ่งเป็นเลขฐานสิบหก

C	ROWIDTOCHAR(rowid)	แปลง rowid ไปเป็นค่าของอักขระ ผลลัพธ์ที่ได้ยาวไม่เกิน 18 อักขระ
C	TO_CHAR(n[,fmt])	แปลง n หรือ d ไปเป็นค่าของอักขระ
C	TO_CHAR(d[,fmt])	ในรูปแบบที่ระบุไว้ตามค่า fmt ถ้าไม่ใช่ fmt แล้ว 1) n ถูกแปลงไปเป็นค่าของอักขระที่ยาวเพียงพอที่จะใส่เลขนัยสำคัญ 2) d ถูกแปลงไปเป็นค่าของอักขระในรูปแบบของวันเดือนปีซึ่งออราเคิลได้กำหนดไว้แล้วคือ 'DD-MON-YY'
D	TO_DATE(char[,fmt])	แปลงวัน เดือนปีจากค่าของอักขระไปเป็นค่าของวันเดือนปี ซึ่ง fmt คือค่าของอักขระที่ระบุรูปแบบของ Char ถ้าไม่ใช่ fmt แล้ว Char ต้องมีรูปแบบวันเดือนปีเป็น 'DD-MON-YY'
D	TO_DATE(n[,fmt])	แปลงตัวเลขเป็นวันเดือนปี
N	TO_NUMBER(char)	แปลงข้อมูลแบบ Char ไปเป็นค่าของตัวเลข

7.5 ฟังก์ชันที่เกี่ยวข้องกับวันเดือนปี

แบบของข้อมูล	ฟังก์ชัน	คำอธิบาย
D	ADD_MONTHS(d,n)	เพิ่มจากวันเดือนปี d ไปอีก n เดือน
D	LAST_DAY(d)	วันเดือนปีของวันสุดท้ายของเดือนที่มีค่า d
N	MONTHS_BETWEEN(d,e)	จำนวนเดือนระหว่างวันเดือนปี d กับ e ถ้า d ล่าสุดกว่า e แล้วผลที่ได้เป็นบวกถ้าไม่เช่นนั้นผลที่ได้เป็นลบ
D	NEW_TIME(d,a,b)	วันเดือนปีและเวลาในแถบเวลา b ถ้าวันเดือนปีและเวลาในแถบเวลา a คือ d ซึ่ง a และ b คือนิพจน์แบบอักขระที่มีความหมายดังต่อไปนี้ AST,ADT Atlantic Standard Time,Daylight Time

BST, BDT Bering Standard Time, Daylight Time
 CST, CDT Central Standard Time, Daylight Time
 EST, EDT Eastern Standard Time, Daylight Time
 GMT Greenwich Mean Time
 HST, HDT Alaska-Hawaii Standard Time, Daylight Time
 MST, MDT Mountain Standard Time, Daylight Time
 NST Newfoundland Standard Time
 PST, PDT Pacific Standard Time, Daylight Time
 YST, YDT Yukon Standard Time, Daylight Time



D NEXT_DAY(d, char) วันเดือนปีที่ เป็นวันแรกของสัปดาห์ซึ่งชื่อ Char. และเท่ากับหรือล่าช้ากว่า d
 N TRUNC(d) ตัดค่าเวลาของวันใน d ที่

7.6 ฟังก์ชันอื่นๆ

แบบของข้อมูล ฟังก์ชัน คำอธิบาย

* DECODE(expr, search1, return1, search2, return2, ... [default])
 ถ้า expr มีค่าเท่ากับค่า search ใดๆ ผลลัพธ์คือค่า return ที่อยู่ถัดมา แต่ถ้าไม่เท่า ผลลัพธ์คือค่า default แต่ถ้าไม่ใช่ค่า default และ expr ก็ไม่เท่ากับ search แล้วผลลัพธ์คือ null
 expr เป็นข้อมูลแบบใดก็ได้ ส่วน search ต้องเป็นข้อมูลแบบเดียวกับ expr และผลลัพธ์ที่ได้จะมีแบบของข้อมูลแบบเดียวกับ return ค่าแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

* `DUMP(expr[,radix[,start-position[,byte]]])`

แสดงค่าของนิพจน์ในรูปแบบภายใน

* `GREATEST(expr,expr,...)`

ค่าที่ใหญ่ที่สุดของลิสต์
สำหรับทุก `expr` หลัง `expr` แรก จะถูก
แปลงไปเป็นแบบของข้อมูลแบบเดียวกับ
`expr` แรกก่อนจะมีการเปรียบเทียบ

* `LEAST(expr,expr,...)` ค่าที่น้อยที่สุดของลิสต์

สำหรับทุก `expr` หลัง `expr` แรก จะถูก
แปลงไปเป็นแบบของข้อมูลแบบเดียวกับ
`expr` แรกก่อนจะมีการเปรียบเทียบ

* `NVL(x,expr)`

ถ้า `x` เป็นนัลส์ ผลลัพธ์คือ `expr` ถ้าไม่
เช่นนั้นผลลัพธ์คือ `x` โดยที่ `x` และ `expr`
เป็นข้อมูลแบบใดก็ได้ ส่วนแบบของข้อมูล
ของผลลัพธ์ต้องเป็นแบบเดียวกับ `x`

* `VSIZE(expr)`

ค่าจำนวนไบนารีของ `expr`

12. คำสั่งเฉพาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต่อ 138 ไปถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12.1 คำสั่งในเอสคิวแอล

คำสั่ง	คำอธิบาย
/*...*/	คำอธิบายจะอยู่ระหว่าง /* และ */
ALTER PARTITION	เพิ่มไฟล์เข้าไปยังพาร์ติชันของฐานข้อมูล
ALTER SPACE	แก้ไขข้อกำหนดสเปซที่ถูกสร้างโดยคำสั่ง CREATE SPACE
ALTER TABLE	เพิ่มคอลัมน์เข้าไปหรือกำหนดคอลัมน์ใหม่ในเทเบิลที่มีอยู่แล้ว
AUDIT	ให้ออราเกิลตรวจสอบการใช้เทเบิล วิว ซินโนนิมหรือระบบ
COMMENT	แทรกคำอธิบายเกี่ยวกับเทเบิลหรือคอลัมน์ในพจนานุกรมข้อมูล
CREATE CLUSTER	สร้างคลัสเตอร์ ซึ่งบรรจุ 2 เทเบิลหรือมากกว่า
CREATE DATABASE LINK	สร้างตัวเชื่อมจากฐานข้อมูลโลกคอล (Loac1) ไปยังชื่อผู้ใช้ในฐานข้อมูลที่อยู่อีกไกล
CREATE INDEX	สร้างอินเด็กซ์สำหรับเทเบิล
CREATE PARTITION	สร้างพาร์ติชันใหม่ในฐานข้อมูล
CREATE SPACE	สร้างข้อกำหนดสเปซซึ่งจะถูกใช้กำหนดคุณสมบัติการแบ่งสเปซของเทเบิล
CREATE SYNONYM	สร้างซินโนนิมสำหรับชื่อเทเบิลหรือชื่อวิว
CREATE TABLE	สร้างเทเบิลและกำหนดคอลัมน์ให้แก่เทเบิลพร้อมทั้งคุณสมบัติอื่นๆ
CREATE VIEW	กำหนดควิวบน 1 เทเบิลหรือมากกว่า หรือ วิวอื่นๆ
DELETE	ลบโรรูว์จากเทเบิลที่ระบุ
DROP	ลบคลัสเตอร์ ตัวเชื่อมฐานข้อมูล เป็นต้น จากฐานข้อมูล
GRANT	สร้าง ID ของผู้ใช้ กำหนดมาลเวริ์ดและยอมให้สิทธิ์พิเศษของออราเกิลแก่ผู้ใช้ หรือยอมให้สิทธิ์พิเศษแก่ผู้ใช้เหนือเทเบิลหรือวิว
INSERT	เพิ่มโรรูว์ใหม่แก่เทเบิลหรือวิว
LOCK TABLE	ล็อกเทเบิลเพื่อให้เข้าถึงมันพร้อมกับผู้ใช้คนอื่นๆ
NOAUDIT	ให้ออราเกิลหยุดการตรวจสอบการใช้เทเบิล วิว ซินโนนิมหรือระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น เมื่อผู้ดูแลเห็นใบเซปวีระเบียนต้นทางการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RENAME	เปลี่ยนชื่อของเทเบิล วิว หรือซันโนนิม
REVOKE	ถอนสิทธิพิเศษฐานข้อมูลหรือสิทธิพิเศษในการเข้าถึง เทเบิลจากผู้ใช้
SELECT	ทำคิวรี หรือคิวรีย่อย, เลือกโรว์ และคอลัมน์จาก 1 เทเบิล หรือมากกว่า
UPDATE	เปลี่ยนแปลงแก้ไขค่าของฟิลด์ (Field) ในเทเบิล
VALIDATE INDEX	ตรวจสอบอินทิกริตี (Integrity) ของอินเด็กซ์บน เทเบิล

12.2 คำสั่งในเอลคิวแอล*พลัส

คำสั่ง	คำอธิบาย
@	รัน (run) ไฟล์คำสั่ง
#	เป็นตัวแสดงว่าจบคำอธิบายซึ่งขึ้นต้นด้วยคำสั่ง DOCUMENT
*	เอกซ์คิวต์ไฟล์คำสั่งของระบบจัดการแบบโฮสต์ (Host) โดยไม่ต้องออกจากเอลคิวแอล*พลัส
/	รันไฟล์คำสั่งในบัฟเฟอร์ของเอลคิวแอล
ACCEPT	พร้อมรับอินพุตที่ผู้ใช้ป้อนและกำหนดเป็นค่าของตัวแปรของผู้ใช้
APPEND	เพิ่มเท็กซ์จนสุดบรรทัดปัจจุบันในบัฟเฟอร์ปัจจุบัน
BREAK	ระบุว่าเหตุการณ์ไหนที่จะหยุดและจะทำอะไรขณะที่หยุด
BTITLE	แสดงชื่อเรื่องที่บรรทัดกลางในแต่ละหน้าของรายงาน
CHANGE	เปลี่ยนแปลงข้อความบางอย่างในบรรทัดที่อยู่ในบัฟเฟอร์ปัจจุบัน
CLEAR	ลบข้อกำหนดของคำสั่ง BREAK ลบเท็กซ์ที่อยู่ในบัฟเฟอร์ปัจจุบัน เป็นต้น
COLUMN	ระบุรูปแบบการแสดงผลของคอลัมน์ และชื่อหรือคำอธิบายคอลัมน์ ในรายงาน
COMMIT	การเปลี่ยนแปลงแก้ไขจะอยู่อย่างถาวร
COMPUTE	คำนวณในกลุ่มของโรว์ที่เลือกมา
CONNECT	ออกจากออรากิล และไปยังชื่อผู้ใช้ตามที่ระบุ
COPY	ถ่ายสำเนาข้อมูลจากเทเบิลในช่วยของฐานข้อมูลหนึ่งของเอลคิวแอลไปยังอีกเทเบิลหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำไปใช้ในเชิงพาณิชย์ การค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMING ทำการวิเคราะห์ประสิทธิภาพของคำสั่งเอคคิวแอล และไฟล์คำสั่ง

TITLE แลคองชื่อ เรื่องที่บรรทัดบนในแต่ละหน้าของเอาท์พุท

UNDEFINE ลมข้อกำหนดเกี่ยวกับตัวแปรที่ผู้ใช้ผู้ใช้สร้างขึ้น

คำสั่งเฉพาะที่กล่าวถึงมานี้เป็นเพียงการอธิบายอย่างสั้นๆ เพื่อให้รู้ว่ามีคำสั่งเฉพาะอะไรบ้าง และคำสั่งเฉพาะนั้นมีไว้เพื่อทำอะไร ถ้าต้องการรายละเอียดเกี่ยวกับคำสั่งเพิ่มเติมก็สามารถค้นหาได้จากหนังสือ "SQL*Plus REFERENCE GUIDE" แต่งโดย Jonathen Sachs, Larry Baer, Derry K&bcenell และ Larry Stevens



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรซี (PRO*CI)

ภาษาเอสคิวแอล อาจกล่าวได้ว่าเป็นภาษาแบบนอนโพรซีดูรอล (Non-procedural) ซึ่งก็คือลักษณะที่ถ้อยแถลง (Statement) เกือบทั้งหมดทำงานโดยอิสระ ไม่ขึ้นกับถ้อยแถลงหน้า และหลังมัน ในขณะที่ภาษาแบบโพรซีดูรอลนั้น มีพื้นฐานอยู่บนรูปแบบที่เป็น ลูป (Loop), บรานช์ (Branch) และ IF/THEN ดังนั้นแม้ว่าภาษาเอสคิวแอล จะมีประสิทธิภาพสูงโดยตัวมันเองก็ตาม แต่ความสามารถนี้ก็ยังคงถูกจำกัดอยู่ เนื่องจากขาดความสามารถต่าง ๆ ที่ภาษาโพรซีดูรอลมี

จากสาเหตุเบื้องต้นนี้ จึงมีการปรับปรุงให้ภาษาเอสคิวแอลสามารถแฝงอยู่ในภาษาที่เป็นโพรซีดูรอล เช่น ซี หรือ ฟอร์แทรน ได้ ซึ่งมีผลก็คือ ทำให้นักโปรแกรมสามารถใช้งานในส่วนที่เป็นข้อดีของทั้งสองภาษาได้

โปรซี ก็เป็นเครื่องมือที่สามารถแปลงโปรแกรมภาษาซีที่ประกอบด้วยถ้อยแถลงของภาษาเอสคิวแอลแทรกอยู่ ให้กลายเป็นโปรแกรมภาษาซีธรรมดาที่สามารถอ้างอิง และใช้งานข้อมูลในออรากเคิลได้ ในฐานะที่เป็นพรีคอมไพเลอร์ (Precompiler) นี้ ภาษาโปรซี จะแปลงประโยค EXEC SQL ... ในไฟล์อินพุต ให้กลายเป็น การเรียกออรากเคิล (Oracle call) ที่เหมาะสมในไฟล์เอาต์พุต จากนั้นเมื่อนำไฟล์เอาต์พุตนี้ไปคอมไพล์ (Compile) โดยใช้คอมไพเลอร์ภาษาซีธรรมดา ร่วมกับการลิงค์ (Link) กับไฟล์ไลบรารี (Library File) ของภาษาซี และออรากเคิล ก็จะได้โปรแกรมเอาต์พุตที่ทำงานได้ตามต้องการ

การใช้งานโดยทั่วไป

1. เขียนโปรแกรมภาษาซีที่มีภาษาเอสคิวแอลแทรกอยู่
2. ใช้ภาษาโปรซี ทำการพรีคอมไพล์โปรแกรมที่เขียน จะได้ไฟล์เอาต์พุต ซึ่งเป็นภาษาซีธรรมดาและแทรกด้วย การเรียกออรากเคิล ออกมา
3. ใช้คอมไพเลอร์ภาษาซีธรรมดา คอมไพล์ไฟล์เอาต์พุต
4. ลิงค์ไฟล์ออบเจ็ค (Object File) ที่ได้ โดยใช้ไลบรารีของภาษาซีและภาษาโปรซีร่วมกัน ก็จะได้โปรแกรมที่ทำงานได้ตามต้องการ

ส่วนวิธีการแทรกภาษาเอสคิวแอลลงในภาษาซี ธรรมดานั้น จะกล่าวถึงในส่วน " การเรียกใช้ เอสคิวแอล จากโปรแกรมระดับสูง "

Compiling and Link Editing

สำหรับ source โปรแกรมภาษา C คำสั่ง cc สามารถทำได้ทั้งการ Compiles และ Link Edits โดยที่ source file จะต้องใช้นามสกุล .c เท่านั้น

การ compilation สามารถทำได้โดยออกคำสั่งที่มีรูปแบบ ดังนี้

* cc filename เช่น * cc mycode.c

ผลลัพธ์ที่ได้จากการ compilation คือ execute file ชื่อ a.out

ในการใช้คำสั่ง cc มีการจัด option ต่าง ๆ เอาไว้ให้เลือกใช้ในการควบคุม การปฏิบัติ ที่พบบ่อย ๆ ดังนี้

-c เป็นการ compilation แบบแยกส่วน ให้ได้ object file เพื่อใช้ในการ link ร่วมกับ file อื่น ๆ

รูปแบบดังนี้ * cc -c mycode.c

จะได้ object file ชื่อ mycode.o

-g เป็นการ compilation แบบให้เพิ่มข้อความพิเศษ ให้กับตัวแปรหรือ statement ใน output file เพื่อใช้ในการ sdb (symbolic debugger)

รูปแบบดังนี้ * cc -g mycode.c

จะได้ execute file ชื่อ a.out

-O เป็นการ compilation ที่ลดขนาดของ object file ให้เล็กที่สุดและ ให้ execute file มีความเร็ว คือทำให้ output file optimization ที่ลด นามใส่ option -O ร่วมกับ -g โดยเด็ดขาด

รูปแบบดังนี้ * cc -O mycode.c

จะได้ execute file ชื่อ a.out

-o เป็นการบอกให้ cc ทราบว่า execute file ที่ได้จากการ compilation ให้เก็บไว้ในชื่อ file ที่เรากำหนด

รูปแบบดังนี้ * cc -o mycode mycode.c

จะได้ execute file ชื่อ mycode

การใช้ Makefile

ในการพัฒนาระบบงานต่อไป ตัว source โปรแกรมจะประกอบไปด้วย Modularity ที่ถูกพัฒนาเพิ่มขึ้นมาเพื่อใช้งานเฉพาะ ต่าง ๆ เพิ่มมากขึ้น ทำให้การ compile ตัวโปรแกรมแต่ละครั้งต้อง compile ทั้งโปรแกรม เพื่อสร้างเป็น execute file ถ้าหากว่าโปรแกรมมีขนาดใหญ่จะทำให้สูญเสียเวลาไปมากกับการ compile โปรแกรมแต่ละครั้ง เช่น ถ้ามีการแก้ไขเพียง modularity เดียวจะยังผลทำให้ต้อง compile โปรแกรมใหม่ทั้งหมด ดังนั้น จึงได้มีการนำเอาหลักของการ compile แยกส่วนมาใช้ โดยในระบบ UNIX ได้แก้ปัญหานี้ด้วยการออกคำสั่ง make เพียงคำสั่งเดียว

make เป็นโปรแกรม (shell commands) ที่ช่วยควบคุมการทำงานการ compile แบบแยกส่วน โดยที่ตัวโปรแกรม makefile จะเก็บเส้นทาง (track) ของคำสั่งที่ใช้สร้าง file และ ความสัมพันธ์ระหว่าง file ต่าง ๆ ไว้ ถ้ามีการสร้าง file หรือ แก้ไข file ใหม่ คำสั่งใน makefile ก็จะต้อง recompiling เฉพาะส่วนที่เพิ่มเข้ามาหรือแก้ไขเท่านั้น โดยไม่มีผลต่อส่วน file อื่น ๆ

การสร้าง makefile สามารถทำได้โดยใช้ Editor สร้างเส้นทางของคำสั่งขึ้นมา แล้วให้ชื่อ file ว่า "makefile" หรือ "Makefile" โดยมีรูปแบบดังตัวอย่าง ต่อไปนี้

ถ้าสมมติว่ามีโปรแกรมย่อย ภาษา C 3 โปรแกรม ดังนี้ xxx.c, yyy.c, zzz.c และ defs.h เป็น header file แล้วเส้นทางของคำสั่งและความสัมพันธ์ระหว่าง file ที่บรรจุใน makefile เป็นดังนี้ โดยที่ defs.h ถูกอ้างถึงโดยโปรแกรม xxx.c และ yyy.c เท่านั้น

```
prog : xxx.o yyy.o zzz.o
      cc -go prog xxx.o yyy.o zzz.o
xxx.o : xxx.c defs.h
      cc -DSCRLB -c -g xxx.c
yyy.o : yyy.c defs.h
      cc -DSCRLB -c -g yyy.c
zzz.o : zzz.c
      cc -DSCRLB -c -g zzz.c
```

โดยสามารถเขียนเป็น diagrams ความสัมพันธ์ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

execute file          prog

object file          xxx.o          yyy.o          zzz.o

source file          xxx.c          yyy.c          zzz.c

include file          defs.h

```

งาน

```

prog : xxx.o yyy.o zzz.o
      cc -go prog xxx.o yyy.o zzz.o

```

คำสั่งข้างบนเป็นส่วนหนึ่งใน makefile โดยส่วนนี้จะสร้างโปรแกรมชื่อ prog ซึ่งเป็น execute file ที่ถูกสร้างขึ้นจากการ compiling และ loading โปรแกรมภาษา C ทั้งสามโปรแกรม คือ xxx.c, yyy.c และ zzz.c โดยที่โปรแกรมทั้ง 3 ได้ถูก compile แยกส่วนมาแล้ว ได้เป็นโปรแกรมชื่อ xxx.o, yyy.o และ zzz.o ตามข้อกำหนดใน makefile ตามลำดับดังนี้

```

xxx.o : xxx.c defs.h
      cc -DSCRLB -c -g xxx.c

```

จะได้ object file ชื่อ ----> xxx.o

```

yyy.o : yyy.c defs.h
      cc -DSCRLB -c -g yyy.c

```

จะได้ object file ชื่อ ----> yyy.o

```

zzz.o : zzz.c
      cc -DSCRLB -c -g zzz.c

```

จะได้ object file ชื่อ ----> zzz.o

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม prog เป็นโปรแกรมที่มีความสัมพันธ์กับ file ที่มีนามสกุล .o ทั้ง 3 ถ้าหากมีการเปลี่ยนแปลงโปรแกรมเพียงโปรแกรมเดียว เช่น yyy.c มันก็จะ recompiled เฉพาะ yyy.c เพื่อสร้าง yyy.o ตัวใหม่แล้วทำการ reload prog ใหม่ และ file xxx.o, yyy.o มีความสัมพันธ์กับ file defs.h โดยที่มันอ้างอิงนิยามที่บรรจุใน defs.h ถ้าหากมีการเปลี่ยนแปลง defs.h จะมีผลทำให้ xxx.c และ yyy.c ถูก recompiled ใหม่ แต่ zzz.c ไม่ต้อง โปรแกรม prog ก็จะถูกสร้างขึ้นมาใหม่จาก xxx.o, yyy.o ตัวใหม่และ zzz.o ตัวเก่า ซึ่งขั้นตอนการทำงานของ compiler แยกส่วนทำโดยอ่านข้อมูลที่กำหนดความสัมพันธ์ของ file ต่าง ๆ แล้วทำการค้นหาในส่วนลึก (depth-first-search) บน diagrams ความสัมพันธ์.

ข้อควรรู้ :

- คำสั่ง make จะเปรียบเทียบวันและเวลาที่เก็บไว้ใน execute file กับใน source file ที่มีการแก้ไขครั้งสุดท้าย ถ้าหากว่าค่าน้อยกว่า คำสั่ง ใน makefile ก็จะถูกปฏิบัติทันที

- ระยะห่างระหว่างจุดเริ่มต้นของบรรทัดกับคำสั่ง cc จะต้องห่างกันอย่างน้อยที่สุด 1 tab ตัวอักษร

- สามารถใช้ macro definitions ใน makefile ได้ เช่น

```
prog : xxx.o yyy.o zzz.o
      cc -go prog xxx.o yyy.o zzz.o
```

สามารถเขียนแทนด้วย

```
obj = xxx.o yyy.o zzz.o
```

```
prog : $(obj)
```

```
cc -go prog $(obj)
```

โดยที่ชื่อ macro obj สามารถที่จะกำหนดเป็นข้อความอื่นได้ตามต้องการ

- สามารถจะแสดงข้อความออกทางจอภาพได้ในระหว่าง execute makefile ได้โดยการใช้คำสั่ง

```
echo ข้อความที่ต้องการแสดงแทรกลงใน makefile
```

การเรียกใช้ SQL จากโปรแกรมภาษาสูง (Embedded SQL)

ส่วนประกอบสำคัญในการเขียนโปรแกรม embedded SQL โดยในที่นี้จะใช้ ภาษา C เป็นภาษาหลัก (Host) มี 3 ส่วน ได้แก่

1. DECLARE Section : เป็นส่วนที่ใช้กำหนด ตัวแปร และ ชนิด ในภาษา C ที่ จะใช้ในการติดต่อกับ SQL
2. INCLUDE SQLCA : เป็นการระบุให้โปรแกรม สามารถเรียกใช้ ส่วนติดต่อกับ ORACLE (SQL Communication Area)
3. CONNECT Stmt. : ใช้ในการติดต่อ หรือ เรียกใช้ ORACLE RDBMS ซึ่ง จะต้องใช้ในโปรแกรม

รายละเอียดของแต่ละส่วน มีดังนี้

1. DECLARE Section : มีรูปแบบในการใช้ดังต่อไปนี้
EXEC SQL BEGIN DECLARE SECTION;
[Variable names and their types]
EXEC SQL END DECLARE SECTION;

สำหรับ ชนิดของตัวแปร (Types) นั้น จะเหมือนกับชนิดของตัวแปรในภาษา C แต่ที่เพิ่มขึ้นมา คือ VARCHAR มีลักษณะเป็นแบบ record เช่น

```
VARCHAR JobDesc[40]; จะแทนได้เป็น  
struct  
{  
    unsigned /* 2 bytes */ short int len;  
    unsigned char arr[40];  
};  
JobDesc;
```

สำหรับ error ที่เกิดขึ้นในส่วน DECLARE นี้ คือ ในกรณีที่ไม่ได้กำหนด ตัวแปร ที่ไว้ในโปรแกรม โดยจะแสดงข้อความ

```
Undeclared host variable <a> at line <b> in file <c>
```

2. INCLUDE SQLCA : มีรูปแบบการใช้

```
EXEC SQL INCLUDE SQLCA;
```

จะทำหน้าที่เปลี่ยน หรือ แทนที่ตัวแปรในโปรแกรม ด้วย ตัวแปรของ ORACLE ในช่วงที่ทำการ precompiled และหน้าที่สำคัญ ของ SQLCA อีกอย่างหนึ่ง นอกเหนือจากการติดต่อกับ ORACLE ก็คือ การแสดง error และ warning ต่างๆ ที่เกิดขึ้นในการปฏิบัติ คำสั่งของ SQL โดยจะแสดงในรูป

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

- sqlca.sqlcode : ถ้า > 0 จะแสดงถึงการกระทำคำสั่ง
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

= ๑ แสดงว่าทำคำสั่งได้สมบูรณ์

< ๑ เกิดการผิดพลาดขึ้น

- sqlca.sqlwarn : จะประกอบด้วย array ของ Flags 8 ตัว ซึ่งแต่ละตัว ก็จะแสดงถึงลักษณะของ warning ที่แตกต่างกันออกไป

นอกจากจะสามารถ เรียกใช้ SQL ได้แล้ว ยังสามารถใช้คำสั่งเฉพาะหรือติดต่อกับ ORACLE ได้โดยตรง โดยใช้คำสั่ง

EXEC SQL INCLUDE ORACA;

3. Connecting to ORACLE : มีรูปแบบการใช้

EXEC SQL CONNECT <oracleid> IDENTIFY BY <oraclepasswd>
หรือ

EXEC SQL CONNECT <oracleid>

โดยที่ oracleid อยู่ในรูป <oracleid>/<oraclepasswd> จะเป็นส่วนที่ต้องใช้ เพื่อให้โปรแกรมสามารถเรียกใช้ ORACLE ได้

ตัวโปรแกรม (Application Body)

เป็นส่วนที่ภาษาหลัก (Host) และ ภาษา embeded รวมกันอยู่ ลักษณะโดยทั่วไปของโปรแกรม คือ

- ภาษาหลัก จะเป็นตัวจัดการ เกี่ยวกับการแสดงผล (display) และ รูปแบบการใช้งานต่างๆ ของโปรแกรม เช่น Menu เป็นต้น
- ภาษา embeded จะทำงานในด้านการจัดการเกี่ยวกับข้อมูล รวมทั้งการเรียกใช้คำสั่งของ SQL และ ORACLE ด้วย ซึ่งการเรียกใช้นั้น จะต้องมีการมี "EXEC SQL" นำหน้าก่อนเสมอ

การถามตอบกับ embeded (Query)

เป็นส่วนหนึ่งในตัวโปรแกรม (Application Body) ซึ่งจะใช้ในการ เรียกข้อมูลมาใช้ หรือ เก็บข้อมูล ต่างๆ

สำหรับคำสั่ง ที่ใช้ใน Query ประกอบด้วย

- | | |
|-------------|----------|
| - SELECT | - INTO |
| - FROM | - WHERE |
| - CONNECT | - UNION |
| - INTERSECT | - MINUS |
| - GROUP BY | - HAVING |

- ORDER BY

สำหรับตัวแปรที่ใช้ใน Query นั้น มาจาก 2 ที่คือ จาก ตาราง (Table) ในภาษา embeded และ จากตัวแปรในภาษาหลัก ซึ่ง ตัวแปรในภาษาหลักที่จะใช้ใน Query จะต้องมีเครื่องหมาย ":" (colon) นำหน้าชื่อตัวแปรเสมอ เช่น

```
SELECT  ENAME,SAL,JOB,MGRNO
INTO    :ename,:osal,:ojob,:omgrno
FROM    EMP
WHERE   :SSNO = 130443139;
```

ลักษณะของ Query มี 2 แบบ คือ

1. Query ที่ให้ผลลัพธ์ออกมาเพียง 1 แถว (Query which return SINGLE ROW only) เป็น Query ที่จะต้องอ้างกับค่าที่มีเพียง 1 แถวในตารางเท่านั้น (Unique Index) ซึ่งถ้าให้ค่ามากกว่า 1 แถว จะแสดง error ออกมา

2. Query ที่ให้ผลลัพธ์มากกว่า 1 แถว (Query which return MULTIPLE ROWS) Query ลักษณะนี้มักใช้กับ การเรียกข้อมูลที่มีเป็นกลุ่มในตาราง ซึ่งเมื่อกระทำ Query นี้แล้ว SQL จะให้ผลลัพธ์ทั้งหมด ออกมาในครั้งเดียว ดังนั้น การใช้ Query แบบนี้ จึงจำเป็นต้องเตรียมพื้นที่ส่วนหนึ่งใน ORACLE หรือ SQL เพื่อที่จะใช้ในการเก็บผลลัพธ์นั้นไว้ แล้วจึงเรียกออกมาใช้ตามที่ต้องการ ซึ่งพื้นที่นั้นจะเรียกว่า Cursor

Cursor มีลักษณะการใช้ดังนี้

- DECLARE CURSOR : เพื่อกำหนดพื้นที่, ชื่อ, และ Query ที่ต้องการ

รูปแบบ : EXEC SQL DECLARE <cursorname> CURSOR FOR [Query];

- OPEN CURSOR : เพื่อเปิดให้สามารถเรียกใช้ Cursor ได้

รูปแบบ : EXEC SQL OPEN <cursorname>;

- FETCH : ให้ Cursor แสดงผลลัพธ์ตัวต่อไป

รูปแบบ : EXEC SQL FETCH <cursorname> INTO <HostVar>;

- CLOSE CURSOR : ยกเลิก Cursor ที่ระบุออกไป

รูปแบบ : EXEC SQL CLOSE <cursorname>;

- CURRENT CURSOR : ให้ใช้ผลครั้งล่าสุดที่มีอยู่ใน Cursor อาจใช้ใน Query

ก็ได้

รูปแบบ : CURRENT OF <cursorname>;

คอมมิต และ โรลแบค (Commit and Rollback)

ในการทำงานของโปรแกรม คำสั่งที่เป็น SQL แต่ละคำสั่ง จะถูก ORACLE มองเป็นส่วนย่อย (logical unit of work) ซึ่งแต่ละส่วนนี้ จะถูกประมวลผลเป็นลำดับขึ้นไปจน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จบ หรือ มีอาจถูกยกเลิกกลางคันก็ได้ สำหรับการยกเลิก unit of work นั้น เกิดได้จาก 2 กรณี คือ

- ผู้ใช้ (user) ยกเลิกเอง
- ระบบ (system) ไม่สามารถทำงานต่อไปได้ เช่น เกิด deadlock ขึ้น และการจบ unit of work มี 2 แบบ คือ
- commit work : เป็นการจบ unit of work โดยให้ทำการเก็บการเปลี่ยนแปลงทั้งหมดที่เกิดขึ้น ไว้ในฐานข้อมูล มีรูปแบบการใช้ คือ

```
EXEC SQL COMMIT WORK [RELEASE];
```

โดยที่ option RELEASE จะเป็นการคืนเนื้อที่ในหน่วยความจำทั้งหมด และออกจากระบบ (log off) ซึ่งจะใช้ในการจบ unit of work สุดท้าย

- rollback work : เป็นการจบ unit of work เช่นกัน แต่จะทำการยกเลิกการแก้ไขข้อมูลทั้งหมด จะใช้ในกรณีที่เกิดการผิดพลาด ในการทำงานของโปรแกรม มีรูปแบบการใช้ลักษณะเดียวกับ commit คือ

```
EXEC SQL ROLLBACK WORK [RELEASE];
```

การแสดงความผิดพลาด (Error and Warning)

หน้าที่สำคัญอีกประการหนึ่งของ SQLCA คือ เป็นส่วนที่จะกระทำเกี่ยวกับการแสดงความผิดพลาดของโปรแกรม หรือ ในส่วนต่างๆ ของ SQL (error handling) โดยที่ลักษณะของ SQLCA จะเป็นโครงสร้าง (structure) ดังนี้

```
struct sqlca {  
    char          sqlcaid [8];  
    long          sqlcabc;  
    long          sqlcode;  
    struct {  
        unsigned short  sqlerrm1;  
        char            sqlerrmc [70];  
    } sqlerrm;  
    char          sqlerrp [8];  
    long          sqlerrd;  
    char          sqlwarn [8];  
    char          sqlext [8];  
};
```

);

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ซึ่งอาจมีลิขสิทธิ์ที่ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบข้อผิดพลาด (Error Detection)

ในการตรวจสอบข้อผิดพลาด (error) จะใช้คำสั่ง WHENEVER ซึ่งจะทำการตรวจที่ SQLCA ทุกๆ ครั้งที่กระทำคำสั่ง SQL มีรูปแบบการใช้ ดังนี้

```
EXEC SQL WHENEVER [ SQLERROR ] [ STOP ] ;  
                   [ SQLWARNING ] [ CONTINUE ] ;  
                   [ NOT FOUND ] [ GOTO stmt-label ] ;
```

โดยที่

- SQLERROR : จะถูก set เมื่อ sqlca.sqlcode เป็นลบ
- SQLWARNING : ----- sqlca.sqlwarn[0] = "w"
- NOT FOUND : ----- sqlca.sqlcode = +1403
(no row found)
- STOP : หยุดการทำงานของโปรแกรม และ rollback
- CONTINUE : ทำงานต่อไป ไม่ว่า sqlca จะเป็นอย่างไร
- GOTO label : ข้ามไปทำที่ stmt-label

โปรซี (PRO*C)

เป็นโปรแกรมที่ใช้ในการแปลงโปรแกรม (source program) ภาษา C ที่มีการ embedded ด้วย SQL ให้เป็นโปรแกรมภาษา C ในรูปที่สามารถติดต่อ และ เรียกใช้ข้อมูลจาก ORACLE RDBMS ได้ PRO*C จะทำการ precompile โปรแกรม โดยที่จะทำการเปลี่ยน EXEC SQL ในตัวโปรแกรมให้เป็นอีกรูปแบบหนึ่ง จากนั้นจึงนำผลที่ได้จากการ precompile ไปทำการ compile, link, และ execute ต่อไป

ขั้นตอนการเขียนโปรแกรม โดยใช้ PRO*C ประกอบด้วย

1. เขียนโปรแกรม embedded ใน C --> source file
2. precompile โดยใช้ PRO*C --> output file
3. compile output program --> object file
4. link-edit object file --> executable file
5. run program --> desired result

หนังสืออ้างอิง

Brian W. Kernighan, Dennis M. Ritchie, "The C Programming Language",
Bell Laboratories, 219 p.

NIXDROF COMPUTER, "Screenlib", 8-2 p. 1987.

Borland, "Turbo C Reference Guide", Borland International, 612 p., 1988

ORACLE, "Pro*C User's Guide", Oracle Corporation, 272 p., 1987

ORACLE, "SQL*Plus User's Guide", Oracle Corporation, 25-12 p., 1987

ORACLE, "SQL*Plus Reference's Guide", Oracle Corporation, 25-12 p., 1988

G. M. NYSSEN, E. D. FALKENBERG, "SQL", Nyssen Data bases, 198 p., 1984

ปัญญานินนธ์นี้จะไม่สามารถสำเร็จลงไปได้ด้วยดี ถ้าปราศจากคำแนะนำจาก อาจารย์ที่ปรึกษา คือ อ.วัชระ ฉัตรวิริยะ ที่ให้ความสนับสนุนในด้านความคิดเห็นในการแก้ไขปัญหาดังๆ ด้วยความเอาใจใส่

ขอบคุณอาจารย์และเจ้าหน้าที่ของภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่าน ที่ได้ให้สัมภาษณ์ในการหาข้อมูลเบื้องต้น ในการวิเคราะห์และออกแบบระบบงาน

ขอบคุณผู้ที่เกี่ยวข้องกับโครงการสารสนเทศทุกท่าน ที่อำนวยความสะดวก เอื้อเฟื้อด้านข้อมูล เอกสาร และให้คำแนะนำในการเขียนโปรแกรม

