

บทที่ 4

ขั้นตอนการวิเคราะห์ และออกแบบการกระจายเพิ่ม

จากบทที่ 3 ได้ทราบถึงปัจจัยที่ใช้ในการพัฒนาการกระจายเพิ่มแล้ว สำหรับบทนี้จะนำไปสู่ขั้นตอนการวิเคราะห์และออกแบบ โดยใช้เครื่องมือที่ได้ศึกษาจากบทที่ 3 โดยใช้โปรแกรมประยุกต์ระบบการจัดการสารสนเทศ สำหรับการบริการวิทยุคมนาคมระบบเซลลูล่าของการสื่อสารแห่งประเทศไทยเป็นกรณีศึกษา ทั้งนี้ระบบงานประยุกต์ดังกล่าวได้ใช้เครื่องเวค และระบบปฏิบัติการวีเอ็มเอส

ก่อนที่จะทำการวิเคราะห์จำเป็นต้องทราบลักษณะงาน และระบบงานประยุกต์ที่ใช้บนเครื่องคอมพิวเตอร์ เพื่อส่งผลให้การวิเคราะห์การกระจายเพิ่มถูกต้อง และแม่นยำมากขึ้น

4.1 ลักษณะระบบงานประยุกต์ที่ใช้เป็นกรณีศึกษา

ระบบงานประยุกต์เป็นการป้อนข้อมูลผ่านทางเทอร์มินัล โดยการปฏิบัติงานของผู้ใช้จะดำเนินงานตั้งแต่ 08.00 ถึง 19.30 น. เพื่อบริการลูกค้า มีส่วนที่ต้องประสานกับชุมสายจะเป็นแบบอัตโนมัติ ดำเนินงาน 24 ชั่วโมงทุกวัน การปฏิบัติงานจะเป็นแบบโต้ตอบ (Interactive) ทั้งหมด

4.2 การศึกษา และวิเคราะห์ระบบงานซอฟต์แวร์ประยุกต์เดิม

ระบบงานที่ใช้เป็นกรณีศึกษาในการวิจัยครั้งนี้เป็นโปรแกรมประยุกต์ที่พัฒนามาเป็นโปรแกรมสำเร็จ (Software Package) ดำเนินงาน (run) บนเครื่องคอมพิวเตอร์ ระบบปฏิบัติการหน่วยความจำเสมือน (Virtual Memory System or VMS) โดยใช้ภาษาคอมพิวเตอร์โคบอล (COBOL language) เป็นโปรแกรมสำหรับพัฒนา เพื่อใช้ดำเนินธุรกิจบริการวิทยุคมนาคมระบบเซลลูล่า (เป็นโปรแกรมประยุกต์ที่ใช้เป็นกรณีศึกษา)

ลักษณะของระบบมี 2 ลักษณะ

1. ประมวลผลข้อมูลที่เกิดจากการดำเนินธุรกิจในแต่ละวัน (Operational System)
2. นำข้อมูลสรุปเป็นข้อสนเทศให้ผู้บริการ (MIS)

การวิเคราะห์ระบบงานในครั้งนี้จะพิจารณาเฉพาะการประมวลผลข้อมูลจากการดำเนินธุรกิจ เนื่องจากเป็นระบบงานที่มีผู้ใช้เข้ามาใช้ระบบเป็นจำนวนมาก ซึ่งการประมวลผลต้องการเวลาตอบสนองที่ดี ส่วนระบบข้อสนเทศสำหรับผู้บริหารนั้น ส่วนใหญ่จะเป็นงานที่ต้องการเพียงผลลัพธ์ที่ต้องใช้เท่านั้น

4.2.1 วิเคราะห์ระบบงานเพื่อพิจารณาแบ่งแยกซอฟต์แวร์ประยุกต์ และข้อมูล เป็นกลุ่ม

เนื่องจากระบบเป็นโปรแกรมสำเร็จจะมีเฉพาะเอกสารวิธีการใช้งาน ฉะนั้นการวิเคราะห์ก็จะศึกษาจากการดำเนินงานประจำวัน และจากเอกสารการใช้งาน วิธีการวิเคราะห์จะไม่เหมือนกับการพัฒนาระบบใหม่ เนื่องจากระบบถูกพัฒนาและใช้งานแล้ว ดังนั้นจึงต้องตรวจโปรแกรมที่ละโปรแกรมเรียกใช้แฟ้มข้อมูลชื่ออะไร (file name) และแฟ้มข้อมูลทั้งหมด ที่โปรแกรมประยุกต์ทั้งระบบมีรวมกันอยู่ในเครื่องคอมพิวเตอร์

การกระจายแฟ้มจะใช้วิธีวิเคราะห์ และพัฒนาแบบแผนโครงสร้าง (Structured Analysis) แต่จะต่างกันตรงการเขียนดีเอพีดี เป็นการทำขั้นตอนย้อนหลังจากระดับบนสุดที่อธิบายระบบได้ละเอียดที่สุด ขึ้นมาเป็นระดับล่างสุดที่รวมโปรแกรมที่หน้าที่เกี่ยวข้องเนื่องกันเข้าด้วยกัน จะใกล้เคียงแผนภาพอรรถาธิบาย (context diagram) มากที่สุด นั่นคือสามารถนำมากำหนดกลุ่มฟังก์ชันการทำงานได้ดีที่สุดโดยมีขั้นตอนดังต่อไปนี้ คือ

1. รวบรวมโปรแกรมทั้งหมดที่มีใช้ในระบบงาน
2. ศึกษาขั้นตอนลำดับในการใช้โปรแกรม (Software routine) เจาะลึกที่ซอฟต์แวร์เฉพาะงาน กำหนดลำดับก่อนหลัง เขียนผังงานระบบ (System flow chart)
3. เขียนแผนภาพกระแสข้อมูล (Data Flow Diagram หรือ DFD) เพื่อดู “กระแส” (Flow) ของข้อมูลจากขบวนการหนึ่งไปอีกขบวนการหนึ่ง
4. ตีกรอบ หรือกำหนดเขต (Boundary) ของ ดีเอพีดี (DFD) เพื่อกำหนดกลุ่มหน้าที่ตามการทำงาน (function) ในแต่ละส่วนของระบบ
5. เมื่อได้กลุ่มการทำงานพิจารณาแฟ้มข้อมูลที่โปรแกรมในกลุ่ม เฉพาะที่เรียกใช้แบบที่มีการเปลี่ยนแปลงข้อมูล แบบเอาพุท (Output) และปรับปรุง (Update)
6. แบ่งงานแต่ละฟังก์ชันการทำงานออกเป็น 3 กลุ่ม (สำหรับติดตั้ง 3 บัพ) โดยพิจารณาแต่ละกลุ่มให้มีการเรียกใช้แฟ้มข้อมูลน้อยที่สุด

4.2.1.1 รวบรวมโปรแกรม และแฟ้มข้อมูลทั้งหมดที่มีใช้ในระบบงาน

โปรแกรมประยุกต์ทั้งหมดจะทำหน้าที่อะไร (ฟังก์ชันการทำงาน) เรียกใช้แฟ้มอะไรบ้าง ลักษณะการเรียกใช้ ความถี่ในการใช้งาน และวิธีการประมวลผล

1. วิธีการประมวลผล

ประมวลผลแบบกลุ่ม (Batch Processing) เป็นการประมวลผลโดยรวบรวมข้อมูลไว้ช่วงระยะเวลาหนึ่งแล้วจึงนำมาประมวลผลทีเดียว

ประมวลผลแบบโต้ตอบ (Interactive Processing) เมื่อมีรายการเปลี่ยนแปลง (Transaction) เกิดขึ้น จะทำการประมวลผลทันที

2. การเรียกใช้แฟ้ม (file usage), input, output, Update)

3. รูปแบบการทำงาน ในแต่ละกระบวนการ จะแบ่งออกได้หลายอย่างดังนี้

3.1 ใช้ในการคำนวณที่ซับซ้อน

3.2 ประมวลผลข้อมูลนำเข้าและต้องการผลลัพธ์ที่มีขนาดใหญ่ และเป็นงานที่เกิดขึ้นซ้ำ ๆ กัน จากข้อมูลขนาดใหญ่

3.3 จัดเรียงข้อมูลเฉพาะข้อมูลที่ต้องการ จากข้อมูลขนาดใหญ่ สรุปยอดรวม แล้วออกรายงาน

3.4 ประมวลผลทันทีที่มีข้อมูลนำเข้า

3.5 สอบถามข้อมูลจากข้อมูลในฐานข้อมูล/เพิ่มข้อมูล

3.6 เปลี่ยนแปลงและปรับปรุงแก้ไขข้อมูล

4. ความถี่ในการใช้งาน

การเรียกใช้โปรแกรมของผู้ใช้เพื่อปฏิบัติงานในหน่วยงานภายในจะมีความบ่อยครั้งแต่ละกระบวนการไม่เหมือนกันขึ้นอยู่กับหน้าที่ดังนี้

4.1. ทำต่อเนื่อง (Continuous)

4.2. รายวัน (Daily)

4.3. รายสัปดาห์ (Weekly)

4.4. รายเดือน (Monthly)

4.5. รายปี (Annually)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

โปรแกรมที่มีใช้งานในระบบทั้งหมด

ชื่อโปรแกรม (Program (P))	หน้าที่ (Function)	รูปแบบการทำงาน (Process Type)	ความถี่ในการดำเนินงาน (Frequency)
1. PRINT_AR	พิมพ์รายงานบัญชีลูกหนี้	ออกรายงาน	รายเดือน
2. AGING_REPORT	พิมพ์รายงานอายุหนี้	ออกรายงาน	รายเดือน
3. INQUIR_AR	สอบถามบัญชีลูกหนี้	สอบถาม	รายวัน
4. PR_REMIND	ออกใบเตือนหนี้	ออกรายงาน	รายเดือน
5. INQUIR_AR_HIS	สอบถามประวัติการชำระหนี้	สอบถาม	รายวัน
6. DISP_DEBT	ออกใบนำส่งเงิน	ออกรายงาน	รายวัน
7. PR_PAGE_REMIND	คัดหมายเลขวิทยุที่ค้างชำระเพื่อโทรเตือนอัตโนมัติ	ออกรายงาน	รายเดือน
8. PR_REMIND_2	ออกใบเตือนหนี้ครั้งที่ 2	ออกรายงาน	รายเดือน
9. SUM_X_ALL	พิมพ์รายงานยอดหนี้ค้างชำระ	ออกรายงาน	รายเดือน
10. BATCH_PAID	บันทึกข้อมูลการชำระเงินส่วนภูมิภาค	ป้อนข้อมูล	รายวัน
11. REP_BATCH_PAID	รายงานการชำระเงินส่วนภูมิภาค	ออกรายงาน	รายวัน
12. UPDATE_PAID	ปรับปรุงข้อมูลการชำระเงินของลูกหนี้ทางบัญชีส่วนภูมิภาค	ปรับปรุงข้อมูล	รายวัน
13. SUM_PAID	รายงานควบคุมการชำระเงินส่วนภูมิภาค	ออกรายงาน	รายเดือน
14. SUM_NONPAID	รายงานควบคุมการค้างชำระส่วนภูมิภาค	ออกรายงาน	รายเดือน
15. NON_PAID	รายงานลูกหนี้การค้างชำระส่วนภูมิภาค	ออกรายงาน	รายเดือน
16. ACCU_CALL	สร้างบิลไดรเวอร์ และคำนวณจำนวนเงินค่าใช้รวมทั้งเดือน	ปรับปรุงข้อมูล	รายเดือน
17. SUMMARY_BILL, SUM_BILL_ACCT	พิมพ์รายงานสรุปใบแจ้งหนี้	ออกรายงาน	รายเดือน
18. ADD_BILL	บันทึกรายการปรับปรุงค่าใช้จ่ายบริการประจำเดือน	ปรับปรุงข้อมูล	รายเดือน
19. PRINT_BILL	พิมพ์ใบแจ้งหนี้, รายงานสรุปใบแจ้งหนี้, และใบฝากส่งไปรษณีย์	ปรับปรุงข้อมูล และพิมพ์รายงาน	รายเดือน

ตารางที่ 4.1 แสดงโปรแกรมที่รวบรวมได้จากโปรแกรมประยุกต์ที่ใช้ในการวิจัย

ชื่อโปรแกรม (Program (P))	หน้าที่ (Function)	รูปแบบการทำงาน (Process Type)	ความถี่ในการดำเนินงาน (Frequency)
20. HOUSE_KEEPING	เตรียมเพิ่มข้อมูลสำหรับเดือนถัดไป	ปรับปรุงข้อมูล	รายเดือน
21.CUST_TEMP_MANI	โปรแกรมเปิดรับรายละเอียดผู้ใช้บริการใหม่	ป้อนข้อมูล	ต่อเนื่อง, รายวัน
22.CUST_DATA_MANI	โปรแกรมเปลี่ยนรายละเอียดข้อมูลผู้ใช้บริการ, ระงับการใช้บริการ, เริ่มการใช้บริการใหม่, ลบรายละเอียดข้อมูลของผู้ใช้บริการ	ป้อนข้อมูล	ต่อเนื่อง, รายวัน
23. CONSISTENCY	ตรวจสอบความถูกต้อง และปรับปรุงข้อมูลผู้ใช้บริการ	ปรับปรุงข้อมูล	ต่อเนื่อง, รายวัน
24. SIM_POST	ส่งข้อมูลเปิดบริการใหม่ หรือเปลี่ยนแปลงไปชุมสาย	ปรับปรุงข้อมูล	ต่อเนื่อง, รายวัน
25.CUSTOMER_INQUIRY	สอบถามรายละเอียดข้อมูลผู้ใช้บริการ สอบถามประวัติการใช้วิทยุคมนาคม, สอบถามรายละเอียดการใช้	สอบถาม	ต่อเนื่อง, รายวัน
26. DAS	ค้นหารายชื่อผู้ใช้วิทยุคมนาคมระบบเซลลูล่า	ป้อนข้อมูล	ต่อเนื่อง, รายวัน
27. INHOUSE_MANI	เพิ่มเติม แก้ไขข้อมูลการทำงานภายใน	ปรับปรุงข้อมูล	เมื่อต้องการ
28. LEVEL_ENTRY	เพิ่มเติม แก้ไขระดับหน้าที่	ปรับปรุงข้อมูล	เมื่อต้องการ
29. OPER_ENTRY	เพิ่มเติม แก้ไขข้อมูลโอเปอเรเตอร์	ปรับปรุงข้อมูล	เมื่อต้องการ
30. SERVICE_CLASS	เพิ่มเติม แก้ไขข้อมูลชนิดการให้บริการ	ปรับปรุงข้อมูล	เมื่อต้องการ
31. MOBILE_DATA_MANI	เปิดรับข้อมูลเครื่องวิทยุคมนาคมใหม่, เปลี่ยนรายละเอียดข้อมูลเครื่องวิทยุคมนาคม, ลบข้อมูลเครื่องวิทยุคมนาคม	ปรับปรุงข้อมูล	ต่อเนื่อง, รายวัน
32. MOB_INQUIRY	สอบถามรายละเอียดข้อมูลเครื่องวิทยุคมนาคม, ประวัติการครอบครองเครื่องวิทยุคมนาคม	สอบถาม	ต่อเนื่อง, รายวัน
33. BATCH_CASH	บันทึกข้อมูลการชำระเงิน	รับข้อมูลนำเข้า	ต่อเนื่อง, รายวัน
34. DATA_INSERT	บันทึกรายการตั้งหนี้เพิ่ม	รับข้อมูลนำเข้า	รายวัน

ตารางที่ 4.1 แสดงโปรแกรมที่รวบรวมได้จากโปรแกรมประยุกต์ที่ใช้ในการวิจัย (ต่อ)

ชื่อโปรแกรม (Program (P))	หน้าที่ (Function)	รูปแบบการทำงาน (Process Type)	ความถี่ในการดำเนินงาน (Frequency)
35. REP_BATCH_CASH	รายงานการชำระเงินเรียงตามหมายเลข BATCH	ออกรายงาน	รายวัน
36. REP_BATCH_ACCT	รายงานการชำระเงินเรียงตามหมายเลขบัญชี	ออกรายงาน	รายวัน
37. UPDATE_PAYMENT	ปรับปรุงข้อมูลการชำระเงินของลูกค้าผ่านทางบัญชี	ปรับปรุงข้อมูล	รายวัน
38. SUM_PAYMENT	รายงานควบคุมการชำระเงิน	ออกรายงาน	รายเดือน
39.SUM_NONPAYMENT	รายงานควบคุมการค้างชำระ	ออกรายงาน	รายเดือน
40. NON_PAYMENT, NON_PAYMENT, NON_GOVERNMENT	รายงานลูกหนี้การค้างชำระเรียงตามหมายเลขบัญชี, ที่ทำการที่ติดตามหนี้, ผู้ใช้ที่เป็นหน่วยงานรัฐบาล	ออกรายงาน	รายเดือน
41. REC_CASH	บันทึกการชำระเงินเพื่อจัดทำ BATCH	รับข้อมูลนำเข้า	รายวัน
42. BANK_MANI	บันทึกข้อมูลรหัสธนาคาร	ปรับปรุงข้อมูล	เมื่อต้องการ
43.CREDIT_BANK_MANI	บันทึกข้อมูลลูกค้าชำระเงินโดยหักบัญชีธนาคาร	ออกรายงาน	รายวัน
44. TRA20R	ยืนยันการหักบัญชีจากธนาคารเพื่อหักล้างบัญชีลูกหนี้	ปรับปรุงข้อมูล	รายเดือน
45.GOVERNMENT_MANI	บันทึกข้อมูลรหัสหน่วยงานราชการ	ปรับปรุงข้อมูล	เมื่อต้องการ
46. CLASS_GRP_MANI	บันทึกข้อมูลประเภทลูกค้า	ปรับปรุงข้อมูล	รายวัน
47.GUARANTEEE_CLASS_MANI	บันทึกข้อมูล การวางหลักประกัน สัญญาของลูกค้า	ปรับปรุงข้อมูล	รายวัน
48. TRACELL	ตั้งห้ราคาใช้บริการในบัญชีลูกหนี้	ปรับปรุงข้อมูล	รายเดือน
49. RATE_DATA_MANI	เปลี่ยนแปลงตารางระยะทาง และช่วงเวลาในกรุงเทพ, ภูมิภาค และระหว่างประเทศ	ปรับปรุงข้อมูล	เมื่อต้องการ
50.TRUNK_TABLE_MANI	เปลี่ยนแปลงตารางที่ตั้งสถานีวิทยุ CELL SITE และ TRUNK GROUP	ปรับปรุงข้อมูล	เมื่อต้องการ
51. RATE_DATA_MANI	เปลี่ยนแปลงอัตราค่าบริการ	ปรับปรุงข้อมูล	เมื่อต้องการ
52. CALLCOST	คำนวณเงินค่าบริการรายวัน	ปรับปรุงข้อมูล	ต่อเนื่อง, รายวัน

ตารางที่ 4.1 แสดงโปรแกรมที่รวบรวมได้จากโปรแกรมประยุกต์ที่ใช้ในการวิจัย (ต่อ)

ชื่อโปรแกรม (Program (P))	หน้าที่ (Function)	รูปแบบการทำงาน (Process Type)	ความถี่ในการดำเนินงาน (Frequency)
53. AMEND_CALL	เปลี่ยนแปลง แก้ไขค่าบริการรายวัน บางรายการ	ปรับปรุงข้อมูล	ต่อเนื่อง, รายวัน
54. CUSTOMER_REP	รายงานข้อมูลหลักผู้ให้บริการแยกตาม กลุ่มตามต้องการ	ออกรายงาน	รายเดือน
55. MOBILE_REP	รายงานข้อมูลหลักเครื่องวิทยุ	ออกรายงาน	รายเดือน
56. MOB_HIS_REP	รายงานประวัติการเปลี่ยนแปลงข้อมูล เครื่องวิทยุ	ออกรายงาน	รายเดือน
57. MOB_CNTL_REP	รายงานการเปลี่ยนผู้ใช้เครื่องวิทยุ	ออกรายงาน	รายเดือน
58. CALL_REP	รายงานการใช้เครื่องวิทยุ	ออกรายงาน	รายวัน
59. TEMP_CUST_LIST	รายงานการแก้ไขข้อมูลผู้ให้บริการ	ออกรายงาน	รายวัน
60. TABLE_REP	รายงานตารางการคิดเงิน	ออกรายงาน	เมื่อต้องการ
61. DPTI	อ่านข้อมูลจากเทปชุมสาย	รับข้อมูลนำเข้า	รายวัน
62. MOBILE_MADE_ENTRY	บันทึกข้อมูลผู้ผลิตเครื่องวิทยุ	ปรับปรุงข้อมูล	เมื่อต้องการ
63. MODEL_ENTRY	บันทึกข้อมูลผู้ผลิตเครื่องวิทยุ	ปรับปรุงข้อมูล	เมื่อต้องการ
64. XPOSTCODE_MANI	บันทึกข้อมูลรหัสไปรษณีย์ ที่ทำการที่ ติดตามหนี้ และที่ทำการที่ทำสัญญา	ปรับปรุงข้อมูล	เมื่อต้องการ
65. FREENUM_MANI	บันทึกข้อมูลหมายเลขปลายทางที่ไม่ คิดค่าบริการ	ปรับปรุงข้อมูล	เมื่อต้องการ
66. EMX_TABLE_MANI	บันทึกข้อมูลรหัสชุมสายวิทยุฯ	ปรับปรุงข้อมูล	เมื่อต้องการ

ตารางที่ 4.1 แสดงโปรแกรมที่รวบรวมได้จากโปรแกรมประยุกต์ที่ใช้ในการวิจัย (ต่อ)

4.2.1.2 เขียนรูปแบบจำลองข้อมูลเชิงตรรกะเป็นแบบจำลองเชิงกายภาพ

(Physical Data Model)

ศึกษาขั้นตอนการปฏิบัติภายในขององค์กรเงื่อนไขที่กำหนด ลำดับก่อนหลัง แล้วนำมาเขียนเป็นผังงานระบบ (System flow chart) โดยศึกษาจากแหล่งข้อมูลต่อไปนี้

แหล่งที่เก็บรายละเอียดมาเพื่อใช้ในการวิเคราะห์

1. คู่มือการใช้โปรแกรมประยุกต์ (Document Gathering)
2. บุคลากรที่ปฏิบัติงานกับโปรแกรมประยุกต์ (People)
3. การสังเกตการณ์ (Observation) ติดตามดูการปฏิบัติงานจริง

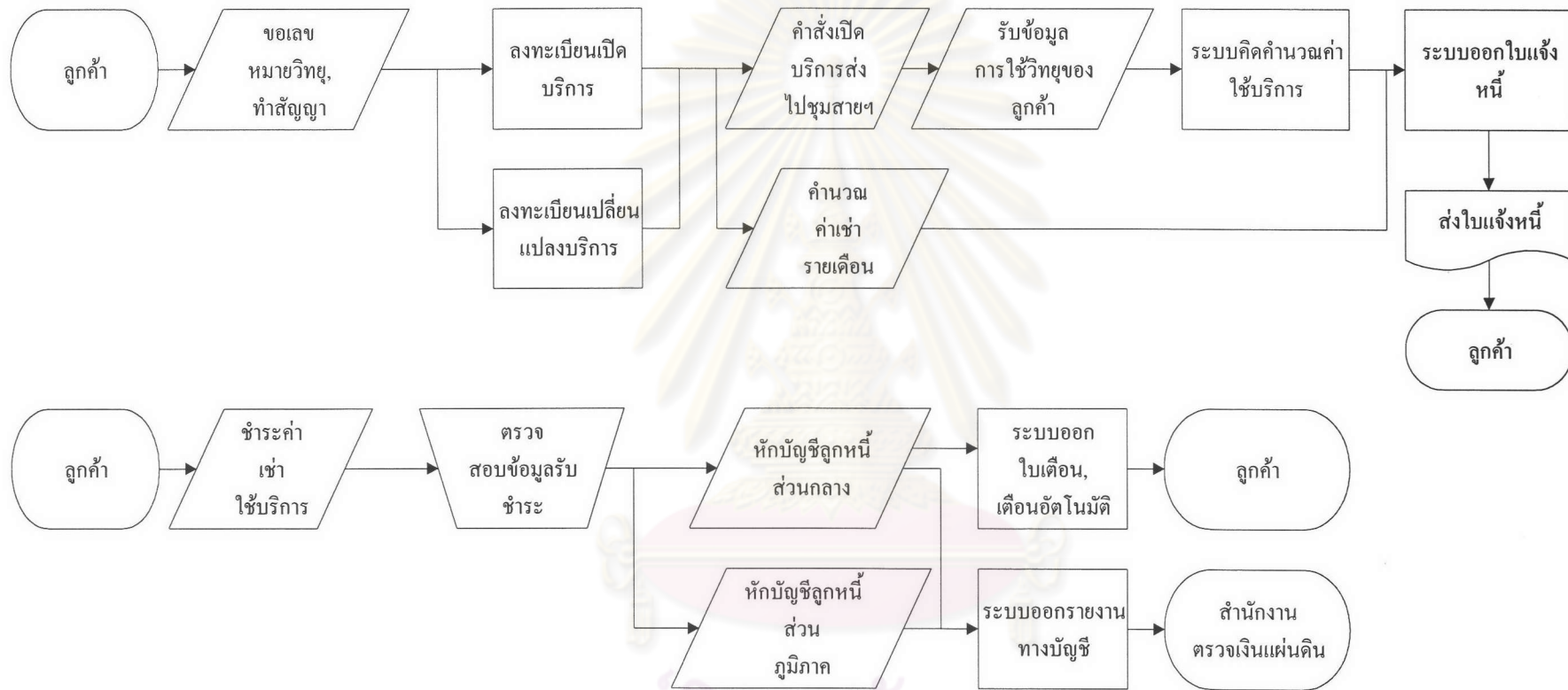
การไหลของข้อมูลในเชิงกายภาพ (physical DFD) แสดงให้เห็นเป็นงาน การหา
ข่าวสารไปตามการไหล (flow) การกระทำกับข่าวสาร คุณสมบัติทางกายภาพประกอบด้วย

1. ชื่อบุคคล
2. ชื่อ จำนวนของแบบฟอร์ม และเอกสาร
3. ชื่อของหน่วยงาน
4. เพิ่มหลักและเพิ่มรายการเปลี่ยนแปลง
5. อุปกรณ์ และวัสดุที่ใช้
6. สถานที่
7. ชื่อของขั้นตอนการปฏิบัติงาน

จากดีเอฟดีข้างต้นนั้นเมื่อรวบรวมข้อมูลส่วนต่างๆ มาประกอบแล้ว สามารถเขียน
ผังงานระบบ (system flowchart) ได้ดังรูปที่ 4.1



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



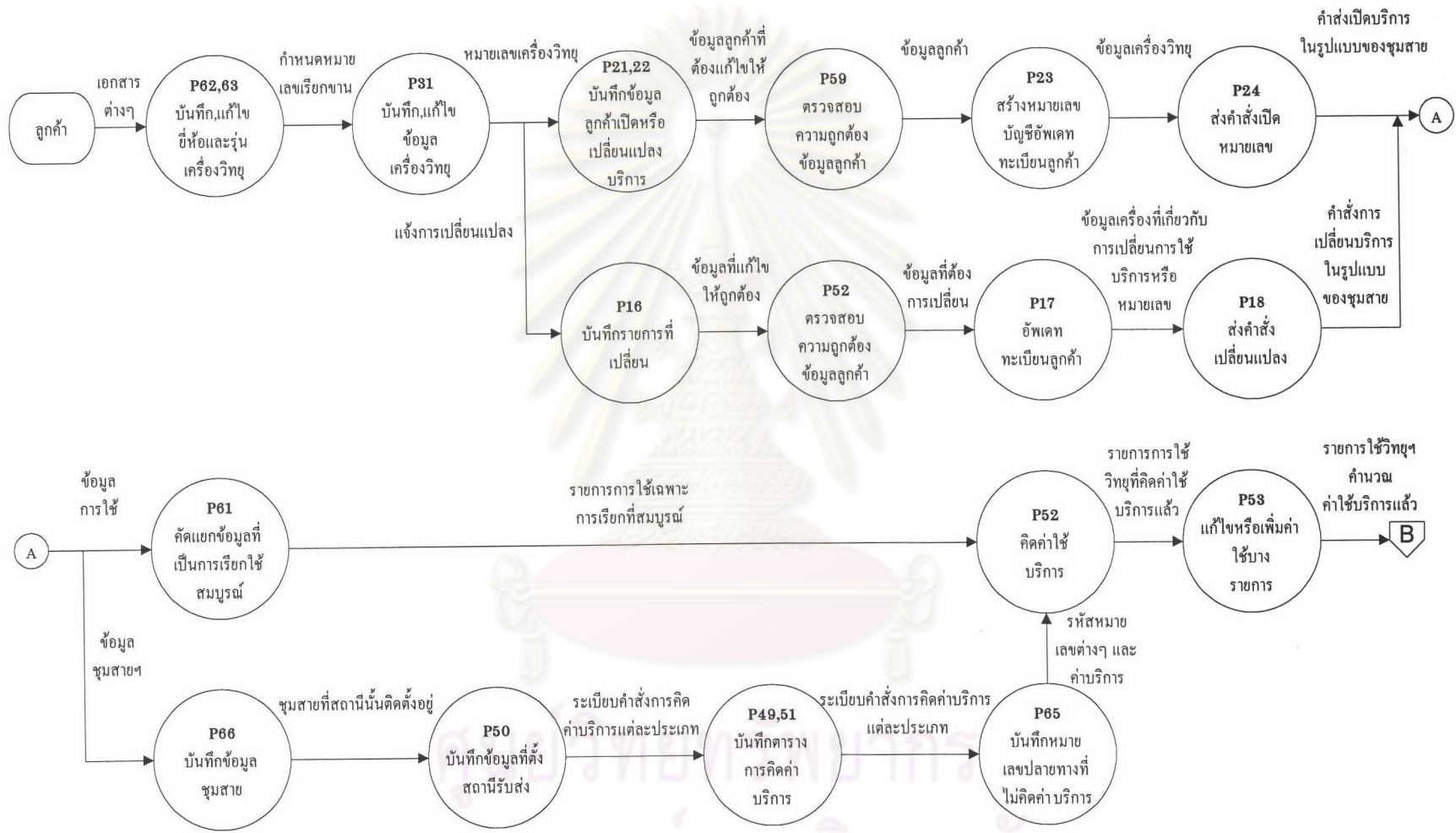
รูปที่ 4.1 ผังงานระบบ (System Flowchart) ของโปรแกรมประยุกต์ที่ใช้ในการวิจัย

4.2.1.3 รูปแบบจำลองข้อมูลเชิงตรรกะ (Logical data model)

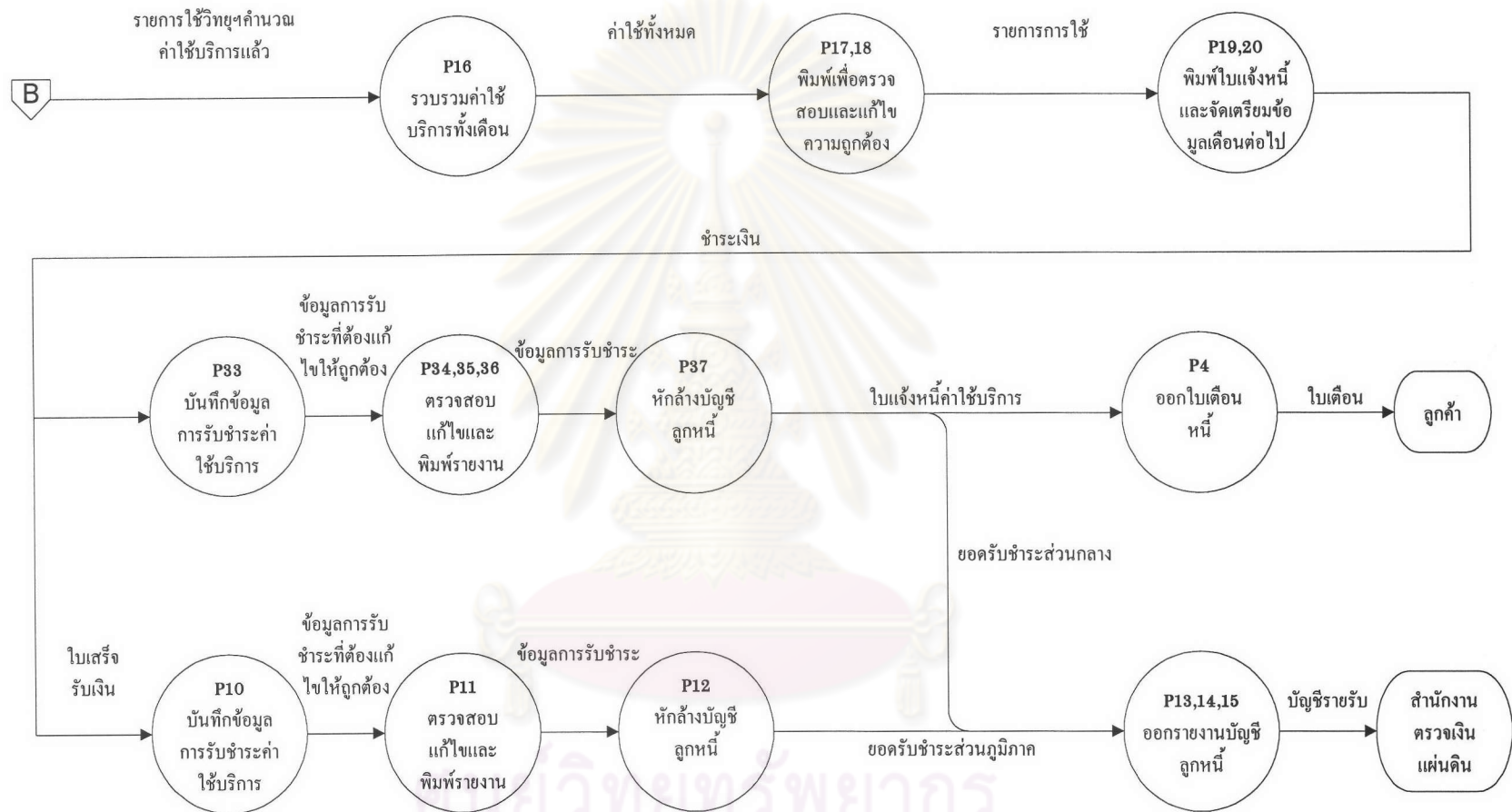
แบบจำลองข้อมูลเชิงตรรกะ (Logical data model) โดยใช้แผนภาพกระแสข้อมูล (Dataflow Diagram) เพื่อใช้ในการอธิบายขั้นตอนการปฏิบัติงาน จุดประสงค์เพื่อแบ่งแยก ฟังก์ชันการทำงาน และจัดกลุ่มได้ง่ายถูกต้องตามหน้าที่การทำงาน โดยเอากระบวนการที่ไม่เปลี่ยนแปลง ข้อมูลออกเช่น กระบวนการใช้ในการสอบถาม, ออกรายงาน เป็นต้น



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.2 แผนภาพกระแสข้อมูล ของโปรแกรมประยุกต์ส่วนที่ 1



รูปที่ 4.3 แผนภาพกระแสข้อมูล ของโปรแกรมประยุกต์ส่วนที่ 2

4.2.1.4 การกำหนดขอบเขตของดีเอฟดีเพื่อแบ่งฟังก์ชันการทำงาน

การแบ่งขอบเขตของดีเอฟดีขึ้นอยู่กับผู้วิเคราะห์ในการตัดสินใจ แต่ก็มีหลักการที่สามารถนำมาใช้ในการพิจารณาในการกำหนดเขตได้จากวิธีการต่อไปนี้

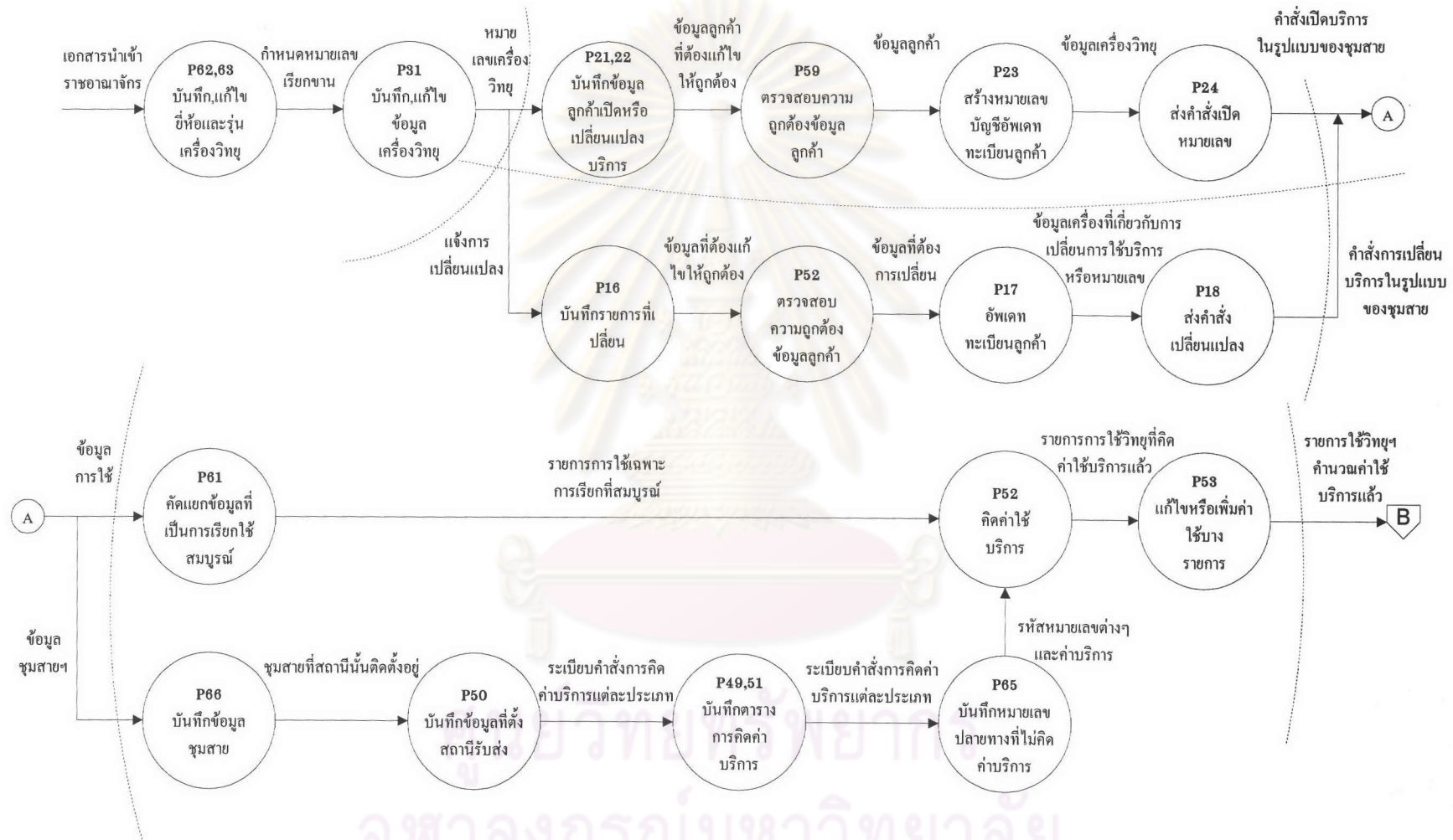
1. ดีกรอบโดยคำนึงถึงงานต่างๆ ภายใน การไหลของข้อมูล และพิจารณาส่วนประกอบต่างๆ ที่ควรจะเป็น โดยใช้วิธีการกำหนดขอบเขต (Boundary) ดังนี้

1. วิเคราะห์แบบทรานฟอร์ม (Transform Analysis)
2. วิเคราะห์แบบทรานแซคชัน (Transaction Analysis)
2. พิจารณาจากความถี่ในการทำงาน จัดอยู่ในกลุ่มเดียวกัน
3. หน้าทีของโปรแกรม
4. การแบ่งหน้าที่ขององค์กร

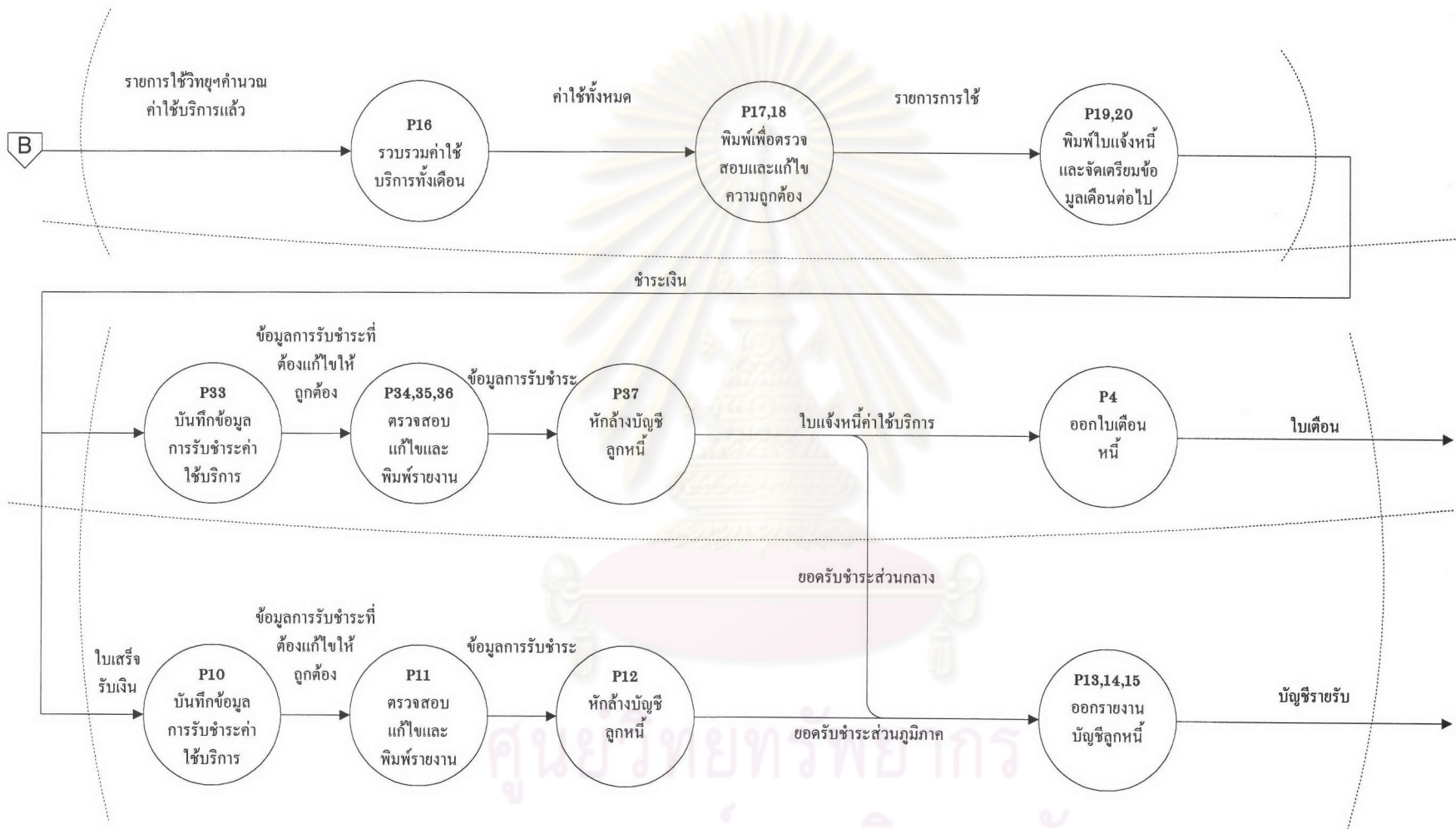
เมื่อแบ่งขอบเขตการทำงานโดยใช้หลักการทั้ง 4 ข้อ มาประกอบในการพิจารณาแล้ว สามารถแบ่งเป็นกลุ่มของโปรแกรมได้ 7 กลุ่ม ซึ่งจะเป็นเขตที่มีฟังก์ชันการทำงานเดียวกัน ได้กำหนดชื่อตามฟังก์ชันหลักของขอบเขตการทำงานนั้นดังนี้

1. ระบบทะเบียนเครื่องวิทยุ
2. ระบบเปิด-ปิดบริการลูกค้า
3. ระบบเปลี่ยนแปลงการใช้บริการ
4. ระบบคิดค่าใช้บริการ
5. ระบบจัดเตรียม และพิมพ์ใบแจ้งหนี้
6. ระบบรับชำระค่าใช้บริการ
7. ระบบบัญชีส่วนกลาง และภูมิภาค

ศูนย์วิทยุทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.4 แบ่งเขตแผนภาพกระแสข้อมูลของโปรแกรมประยุกต์เป็นกลุ่ม ส่วนที่ 1



รูปที่ 4.5 แบ่งเขตแผนภาพกระแสข้อมูลของโปรแกรมประยุกต์เป็นกลุ่ม ส่วนที่ 2

4.2.1.5. รวบรวมโปรแกรมแต่ละฟังก์ชันการทำงานที่ได้แบ่งขอบเขตมาแล้ว

การรวมเพิ่มข้อมูลทั้งหมดในแต่ละกลุ่มงาน ซึ่งทั้งเขตจะเป็นเขตที่มีฟังก์ชันการทำงานเดียวกัน เพื่อกำหนดเพิ่มข้อมูลที่เป็นเพิ่มหลัก

ระบบเพิ่มข้อมูลที่มีใช้งานในระบบทั้งหมด

1. เพิ่มทะเบียนผู้ใช้บริการชื่อ CUSTOMER
2. เพิ่มประวัติการเปลี่ยนแปลงข้อมูลของผู้ใช้บริการ ชื่อ CUS_HISTORY
3. เพิ่มชนิดที่มีให้บริการและอัตราชื่อ SERVICE_CLASS
4. เพิ่มผู้ใช้บริการแยกตามกลุ่มชื่อ CLASS_GRP
5. เพิ่มรหัสส่วนราชการแบ่งตามกระทรวงชื่อ GOVERNMENT
6. เพิ่มทะเบียนเครื่องวิทยุ ชื่อ MOBILE
7. เพิ่มประวัติการเปลี่ยนแปลงเครื่องวิทยุ ชื่อ MOB_HISTORY
8. เพิ่มเก็บหมายเลขบัญชีช่วงตั้งแต่เปิดบริการถึงปิดบริการชื่อ MOB_CONTROL
9. เพิ่มเก็บรายละเอียดผู้ผลิตเครื่องวิทยุ ชื่อ MOB_MADE
10. เพิ่มเก็บหมายเลขรุ่น และชนิดของเครื่องวิทยุ ชื่อ MODEL
11. เพิ่มบัญชีลูกหนี้ชื่อ TRANSACTION
12. เพิ่มประวัติการชำระค่าบริการชื่อ TRANS_HIS
13. เพิ่มรับชำระค่าเช่า/ใช้บริการชื่อ CASH_cc_mddd
14. เพิ่มเก็บที่ทำการที่ติดตามหนี้ตามที่อยู่ของผู้ใช้บริการชื่อ POSTCODE
15. เพิ่มรหัสธนาคารชื่อ BANK
16. เพิ่มเก็บหมายเลขบัญชีผู้ใช้บริการที่ชำระค่าใช้บริการผ่านธนาคารชื่อ

CREDITBANK

17. เพิ่มเก็บรหัสวิธีการที่ผู้ใช้บริการไปชำระค่าบริการชื่อ RECEIVE_TYPE
18. เพิ่มเก็บรหัสและที่ตั้งของชุมสายวิทยุชื่อ EMX_TABLE
19. เพิ่มเก็บรหัสและที่ตั้งของสถานีรับส่งวิทยุชื่อ TRUNK_TABLE
20. เพิ่มเก็บตาราง รหัสหมายเลขโทรศัพท์ปลายทาง ในกรุงเทพฯ ชื่อ

LOCAL_SWITCH

21. เพิ่มเก็บตารางช่วงเวลาที่ยึดราคาค่าบริการลด/เพิ่ม ที่หมายเลขโทรศัพท์ปลายทางในกรุงเทพฯ ชื่อ LOCAL_TIME
22. เพิ่มเก็บตารางรหัสหมายเลขโทรศัพท์ปลายทางในกรุงเทพฯ แบ่งเป็นกลุ่มที่มีอัตราเท่ากัน ชื่อ LOCAL_ZONE

23. เพิ่มเก็บตารางรหัสหมายเลขโทรศัพท์ปลายทางทั้งประเทศชื่อ DOM_SWITCH
24. เพิ่มเก็บตารางรหัสหมายเลขวิทยุคมนาคมระบบเซลลูล่าปลายทางที่เป็นของผู้ให้บริการรายอื่น ที่มีให้บริการชื่อ DOM_SW_MTX
25. เพิ่มเก็บตารางช่วงเวลาที่ยึดราคาค่าบริการลด/เพิ่มที่หมายเลขโทรศัพท์ปลายทางทั้งประเทศ ชื่อ DOM_TIME
26. เพิ่มเก็บตารางรหัสหมายเลขโทรศัพท์ปลายทางทั้งประเทศแบ่งเป็นกลุ่มที่มีอัตราเท่ากัน ชื่อ DOM_ZONE
27. เพิ่มเก็บตารางรหัสหมายเลขโทรศัพท์ปลายทางระหว่างประเทศชื่อ INT_TABLE
28. เพิ่มเก็บตารางช่วงเวลาที่ยึดราคาค่าบริการลด/เพิ่มที่หมายเลขโทรศัพท์ปลายทางระหว่างประเทศ ชื่อ INT_TIME_TBL
29. เพิ่มอัตราค่าใช้บริการ ชื่อ SALE_PLAN_TBL
30. เพิ่มเก็บรหัสหมายเลขปลายทางที่ไม่คิดค่าบริการ ชื่อ FREE_NUM
31. เพิ่มควบคุมตำแหน่งของการบันทึกข้อมูลการเรียกใช้วิทยุคมนาคมชื่อ CALL_CONTROL
32. เพิ่มข้อมูลการเรียกใช้วิทยุคมนาคม ที่คิดค่าบริการแล้ว ชื่อ LOCAL, DOMESTIC, INTERNATIONAL CALLmm
33. เพิ่มตั้งค่าเช่ารายเดือน ชื่อ BILL_DETAIL
34. เพิ่มเก็บรหัสเพื่อควบคุมลำดับขั้นตอนของระบบเพื่อให้ระบบดำเนินไปอย่างถูกต้อง ชื่อ IN_HOUSE
35. เพิ่มข้อมูลการเรียกใช้วิทยุคมนาคม ชื่อ CSCLT
36. เพิ่มบัญชีลูกหนี้ส่วนภูมิภาคชื่อ PAID
37. เพิ่มประเภทการชำระเงินชื่อ PAID_TYPE
38. เพิ่มระดับที่อนุญาตให้ผู้ปฏิบัติงานเข้าใช้โปรแกรมในระบบได้ ชื่อ PROG_LEVEL
39. เพิ่มรายละเอียดผู้ปฏิบัติงานและระดับที่ผ่านเข้าใช้โปรแกรมในระบบชื่อ OPER_ENG
40. เพิ่มใช้งานชั่วคราวชื่อ Temporary, Driver
41. เพิ่มพิมพ์รายงานชื่อ Report

ชนิดของแฟ้มข้อมูล ในวงการธุรกิจ และองค์กรต่าง ๆ ไม่ว่าจะใช้คอมพิวเตอร์หรือไม่ ต้องมีแฟ้มเก็บเอกสารข้อมูลต่าง ๆ หลายชนิดรวบรวมไว้ในองค์กรธุรกิจ ชนิดของแฟ้มข้อมูลมีอยู่หลายลักษณะด้วยกันดังนี้

1. แฟ้มหลัก (Master File) เป็นแฟ้มข้อมูลที่ประกอบด้วยข้อมูลสำคัญของระบบที่จะต้องนำมาใช้อยู่เสมอมีหลายรูปแบบ และต้องปรับปรุงให้ทันสมัยตลอดเวลาที่ยังใช้งาน ในระบบหนึ่งอาจมี แฟ้มหลัก มากกว่า 1 แฟ้มก็ได้

2. แฟ้มรายการเปลี่ยนแปลง (Transaction File) เป็นแฟ้มนำเข้าของระบบงานที่นำเข้ามาเก็บข้อมูลสะสมไว้จากการดำเนินธุรกิจหรือการประมวลผลแบบ Batch แฟ้มนี้จะสัมพันธ์กับแฟ้มหลักคือ เข้าไปแก้ไข และปรับปรุงข้อมูลในแฟ้มนั้น

3. แฟ้มอ้างอิงรายการเปลี่ยนแปลง (Transaction Log File) คล้ายกับแฟ้มข้อมูลรายการเปลี่ยนแปลง แต่ข้อมูลในแฟ้มนี้ได้ผ่านการประมวลผลในแบบ Interactive แล้ว

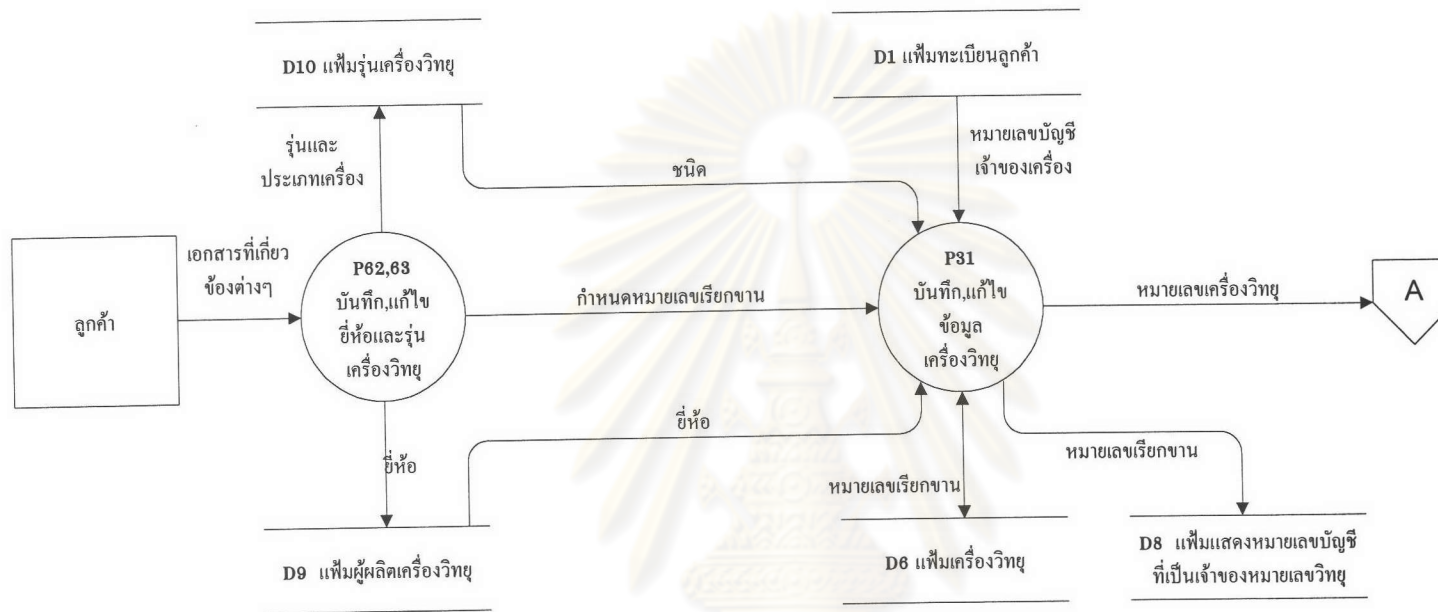
4. แฟ้มสำรอง (Backup File) เป็นแฟ้มสำรองข้อมูล ของแฟ้มต่างๆ ไปในระบบ

5. แฟ้มตาราง (Table File) ใช้เก็บตารางเปรียบเทียบค่าเอาไว้ จะนำมาใช้งานเมื่อต้องการ

6. แฟ้มประวัติ (History File) เป็นแฟ้มข้อมูลหลักที่ไม่ได้ใช้งานแล้ว แต่เก็บเอาไว้เพื่อผลทางสถิติ การอ้างอิง หรือเป็นแฟ้มที่เก็บข้อมูลที่ได้ถูกลบออกมาจากแฟ้มหลัก

7. แฟ้มเก็บข้อผิดพลาด (Error File) ใช้เก็บเฉพาะข้อมูลที่ไม่ถูกต้อง เพื่อนำไปแก้ไขต่อไป เช่น การประมวลผลแบบใช้แฟ้มรายการเปลี่ยนแปลง จะต้องผ่านการตรวจสอบข้อผิดพลาดเสียก่อน ข้อมูลที่ไม่ถูกต้องจะถูกบันทึกลงในแฟ้มชนิดนี้แล้วนำไปแก้ไขก่อนเอากลับเข้ามาประมวลผลใหม่

8. แฟ้มรายงาน (Report File) สร้างขึ้นเพื่อนำมาออกรายงาน เป็นประโยชน์ในการแบ่งใช้อุปกรณ์เช่น ในหลักการของอุปกรณ์เสมือน โดยที่ไม่ต้องพิมพ์ทันที สามารถเก็บผลลัพธ์ไว้ตรวจสอบก่อน จึงพิมพ์ออกเครื่องพิมพ์จริง และช่วยให้ไม่ต้องเสียเวลาในการประมวลผลใหม่ ในระหว่างที่พิมพ์เกิดขัดข้อง หรือต้องการพิมพ์หลายชุด



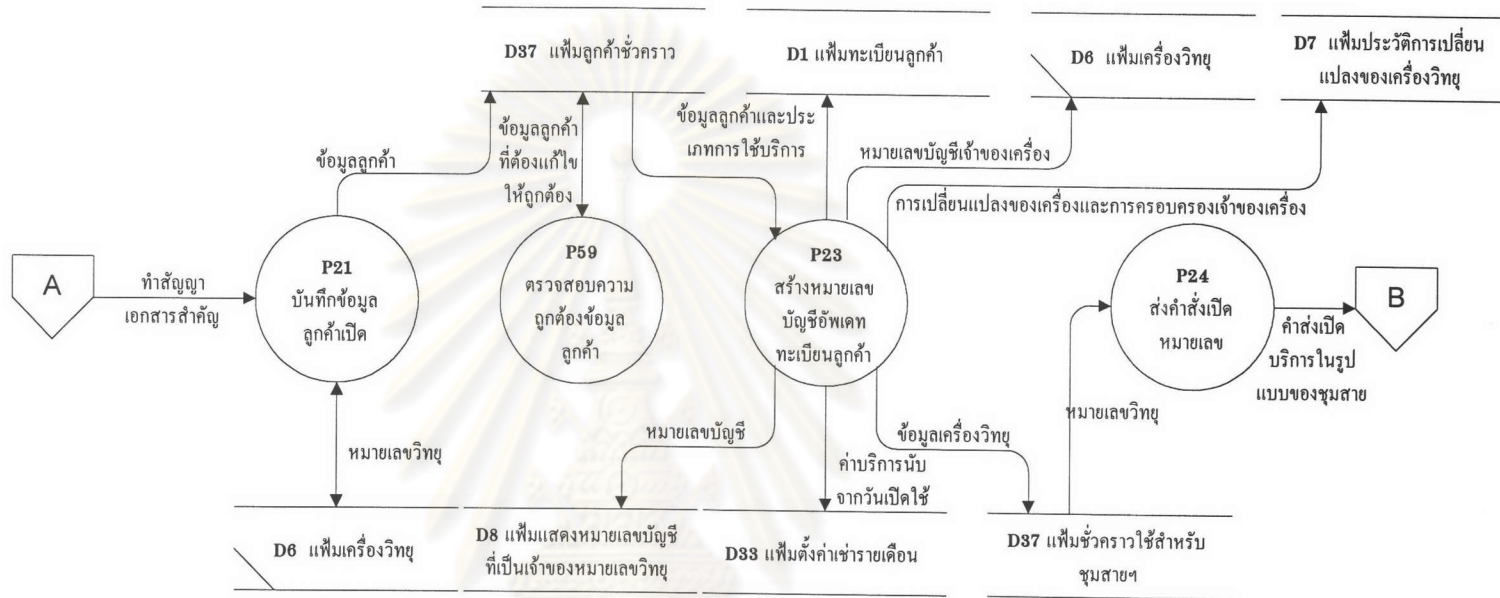
Process	Frequency	Type	Program
P62	ครั้งแรก	Interactive Online	MOBILE_MADE_ENTRY
P63	ครั้งแรก	Interactive Online	MODEL_ENTRY

Process	Frequency	Type	Program
P31	ตลอดวัน	Interactive Online	MOBILE_DATA_MANI

Data Store	ชื่อแฟ้ม	การแก้ไขแฟ้ม
D9	MOB_MADE	Update
D10	MODEL	Update

Data Store	ชื่อแฟ้ม	การแก้ไขแฟ้ม
D1	CUSTOMER	Input
D6	MOBILE	Update
D8	MOB_CONTROL	Output
D9	MOB_MADE	Input
D10	MODEL	Input

รูปที่ 4.6 แสดงแผนภาพกระแสข้อมูล และที่เก็บข้อมูลของโปรแกรมประยุกต์ ส่วนที่ 1



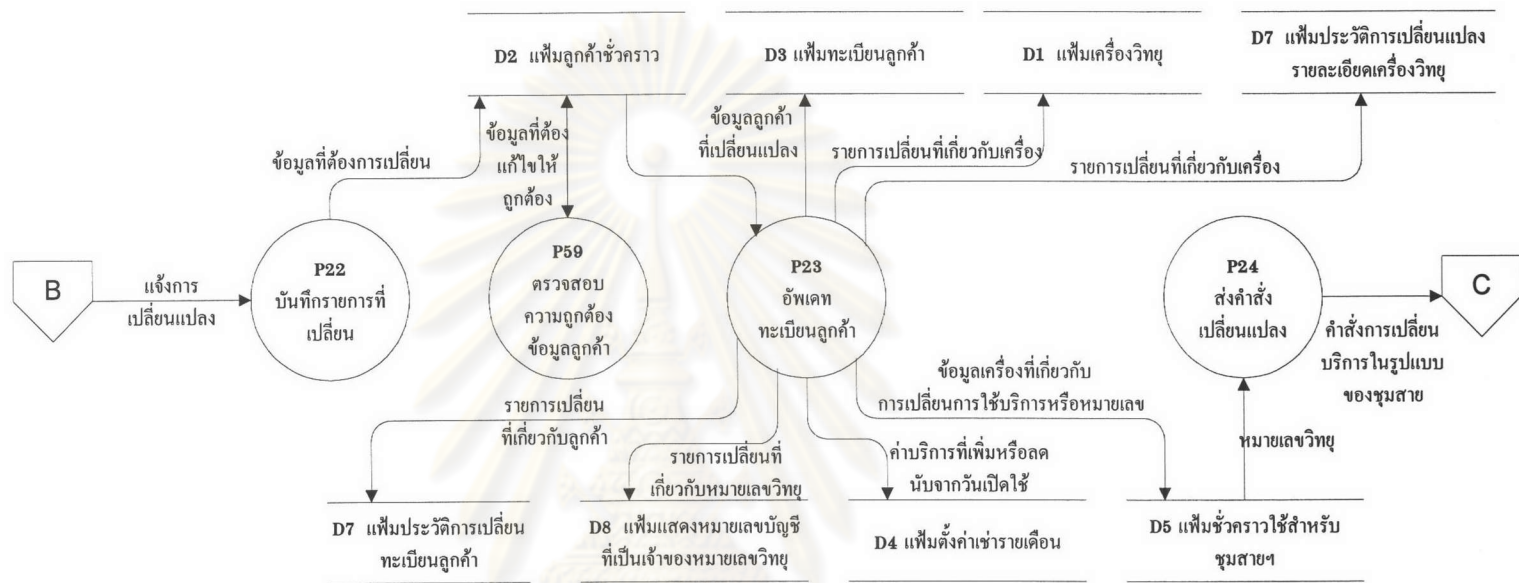
Process	Frequency	Type	Program
P21	ตลอดวัน	Interactive Online	CUST_TEMP_MANI
P59	ตลอดวัน	Interactive Online	TEMP_CUST_LIST

Process	Frequency	Type	Program
P23	ตลอดวัน	Interactive Online	CONSISTENCY
P24	ตลอดวัน	Interactive Online	SIM_POST

Data Store	ชื่อเพิ่ม	การแก้ไขเพิ่ม
D6	MOBILE	Update
D37	TEMP_CUST	Output

Data Store	ชื่อเพิ่ม	การแก้ไขเพิ่ม
D1	CUSTOMER	Update
D6	MOBILE	Input
D7	MOB_HISTORY	Output
D8	MOB_CONTROL	Update
D33	BILL_DETAIL	Output
D37	SIM_DRIVER	Output
D37	TEMP_CUST	Input

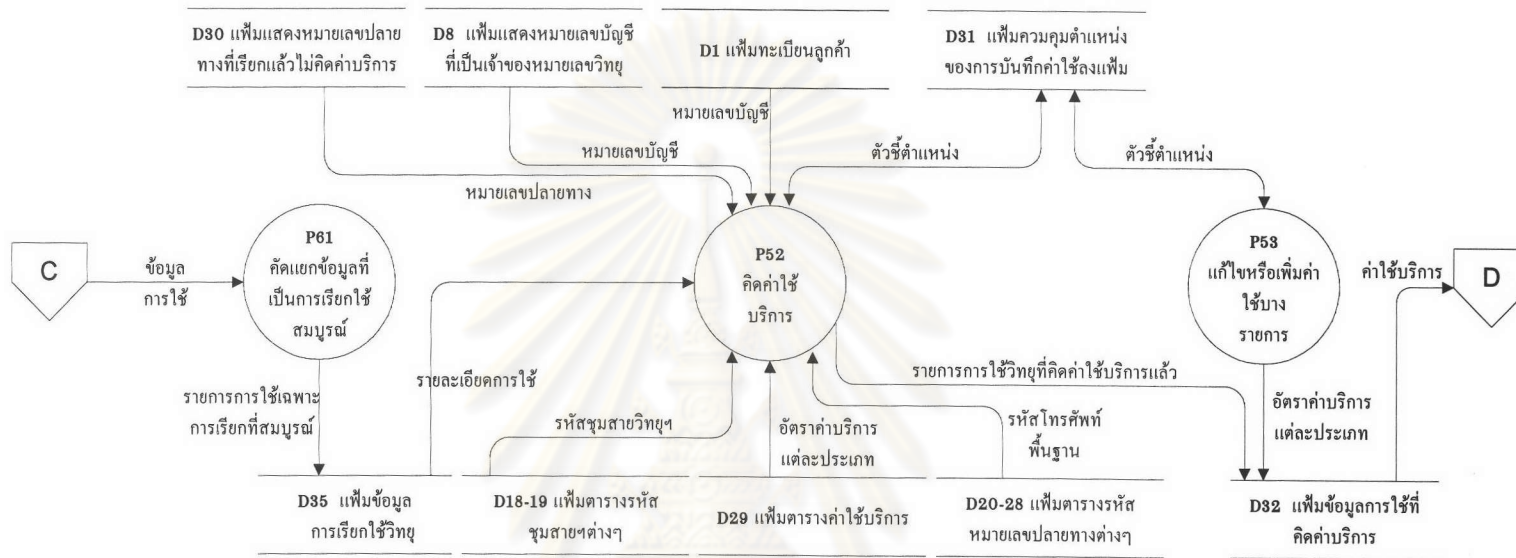
รูปที่ 4.7 แสดงแผนภาพกระแสข้อมูล และที่เก็บข้อมูลของโปรแกรมประยุกต์ ส่วนที่ 2



Process	Frequency	Type	Program
P22	ตลอดวัน	Interactive Online	CUST_DATA_MANI
P59	ตลอดวัน	Interactive Online	TEMP_CUST_LIST
P23	ตลอดวัน	Interactive Online	CONSISTENCY
P24	ตลอดวัน	Interactive Online	SIM_POST

Data Store	ชื่อแฟ้ม	การเข้าใช้แฟ้ม
D1	CUSTOMER	Update
D2	CUS_HISTORY	Update
D6	MOBILE	Input
D7	MOB_HISTORY	Output
D8	MOB_CONTROL	Update
D33	BILL_DETAIL	Output
D37	SIM_DRIVER	Output
D37	TEMP_CUST	Input

รูปที่ 4.8 แสดงแผนภาพกระแสข้อมูล และที่เก็บข้อมูลของโปรแกรมประยุกต์ ส่วนที่ 3



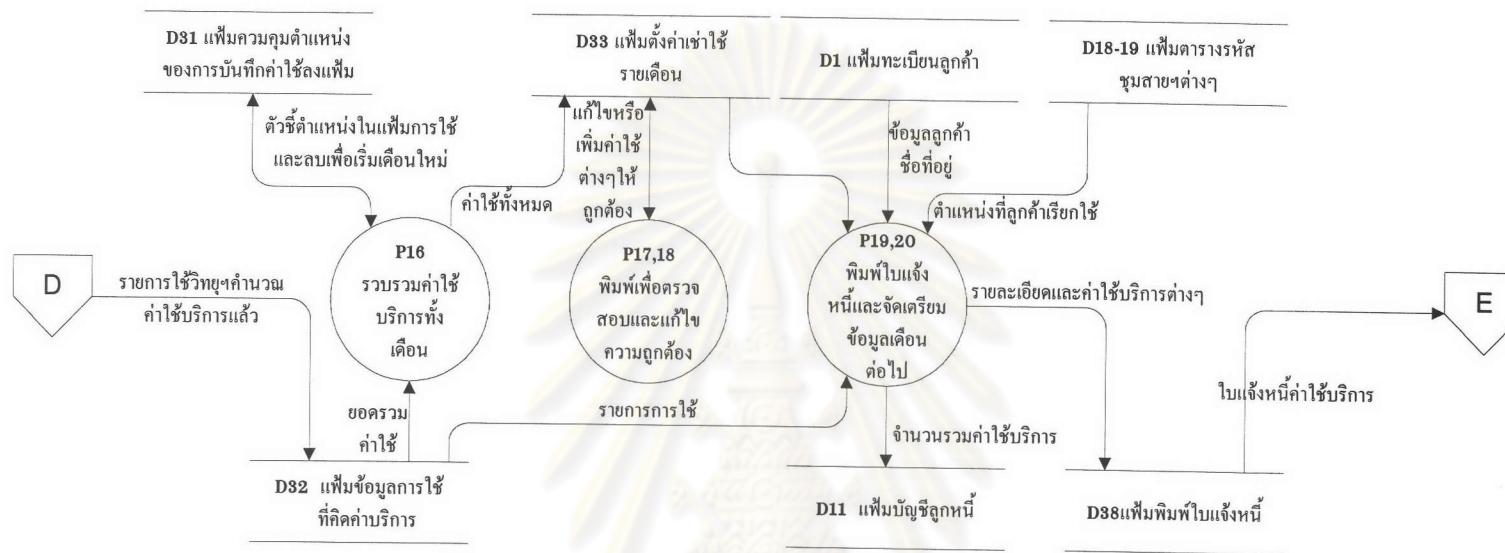
Process	Frequency	Type	Program
P61	ตลอดวัน	Batch Online	DPTI

Data Store	ชื่อเพิ่ม	การเข้าใช้เพิ่ม
D35	CSCLT	Output

Process	Frequency	Type	Program
P52	ตลอดวัน	Batch Online	CALLCOST
P53	สิ้นวัน	Interactive Online	AMEND_CALL

Data Store	ชื่อเพิ่ม	การเข้าใช้เพิ่ม
D1	CUSTOMER	Input
D8	MOB_CONTROL	Input
D31	CALL_CONTROL	Update
D35	CSCLT	Input
D18-19	EMX, TRUNK-TABLE	Input
D20-28	LOCAL, DOMESTIC, INTER-TABLE	Input
D29	SALE_PLAN_TBL	Input
D30	FREE_NUM	Input
D32	LOCAL, DOMESTIC, INTER-CALL	Update

รูปที่ 4.9 แสดงแผนภาพกระแสข้อมูล และที่เก็บข้อมูลของโปรแกรมประยุกต์ ส่วนที่ 4



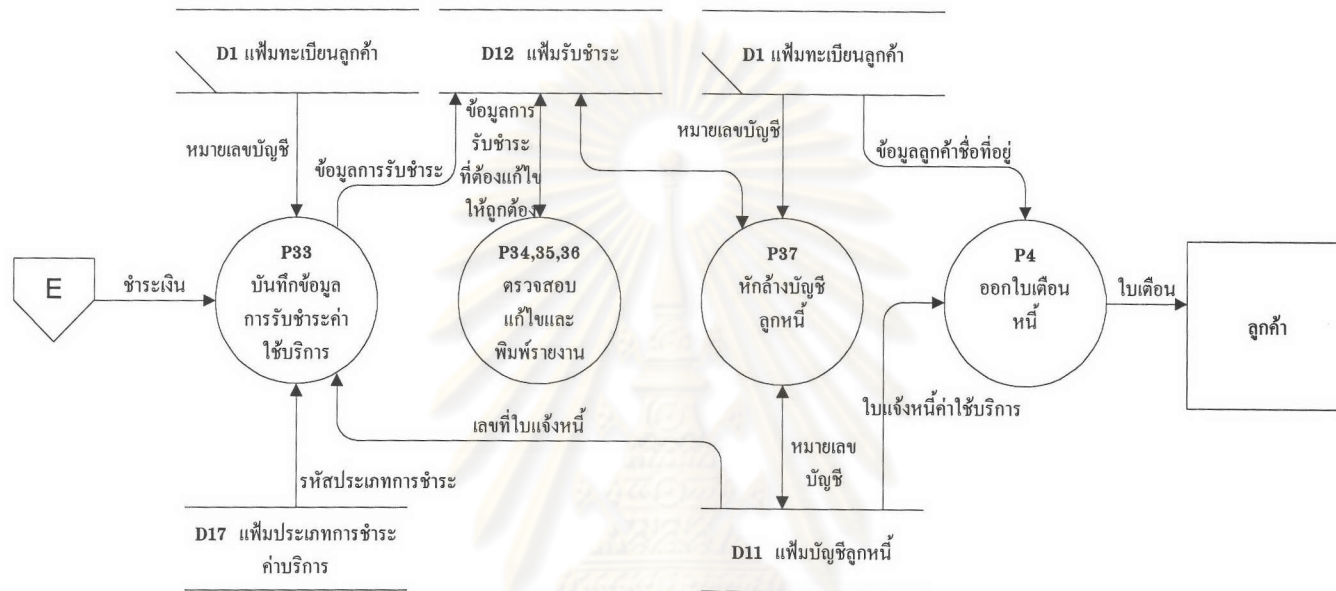
Process	Frequency	Type	Program
P16	สิ้นเดือน	Batch Online	ACCU_CALL
P17	สิ้นเดือน	Batch Online	SUMMARY_BILL
P18	สิ้นเดือน	Batch Online	ADD_BILL

Process	Frequency	Type	Program
P19	สิ้นเดือน	Interactive Online	PRINT_BILL
P20	สิ้นเดือน	Interactive Online	HOUSE_KEEPING

Data Store	ชื่อแฟ้ม	การเข้าใช้แฟ้ม
D1	CUSTOMER	Input
D31	CALL_CONTROL	Update
D32	LOCAL,DOMESTIC,INTER-CALL	Input
D33	BILL_DETAIL	Update
D41	REPORT	Output

Data Store	ชื่อแฟ้ม	การเข้าใช้แฟ้ม
D1	CUSTOMER	Input
D8	MOB_CONTROL	Input
D31	CALL_CONTROL	Update
D35	TRANSACTION	Output
D36	PAID	Output
D18-19	EMX, TRUNK-TABLE	Input
D32	LOCAL,DOMESTIC,INTER-CALL	Input
D33	BILL_DETAIL	Input
D41	REPORT	Output

รูปที่ 4.10 แสดงแผนภาพกระแสข้อมูล และที่เก็บข้อมูลของโปรแกรมประยุกต์ ส่วนที่ 5

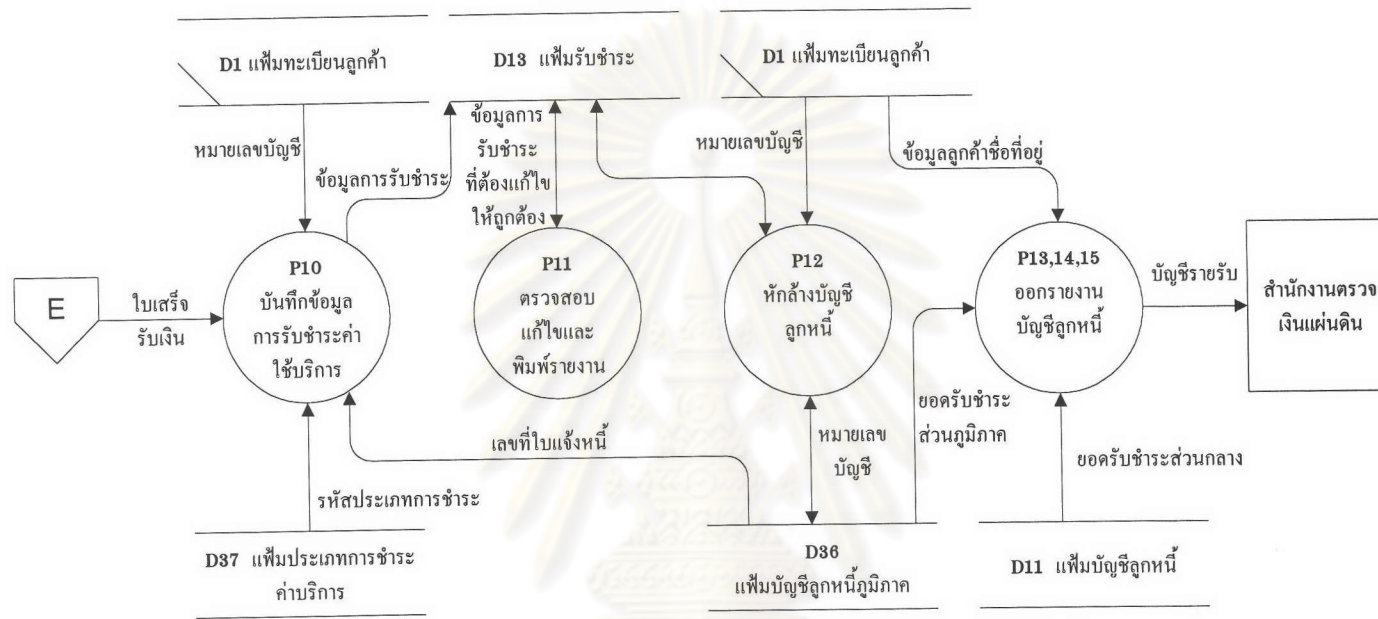


Process	Frequency	Type	Program
P33	ตลอดวัน	Interactive Online	BATCH_CASH
P34	สิ้นเดือน	Interactive Online	DATA_INSERT
P35	ตลอดวัน	Batch Online	REP_BATCH_CASH
P36	ตลอดวัน	Batch Online	REP_BATCH_ACCT
P37	ตลอดวัน	Batch Online	UPDATE_PAYMENT

Process	Frequency	Type	Program
P4	สิ้นเดือน	Batch Online	PR_REMIND

Data Store	ชื่อเพิ่ม	การเข้าใช้เพิ่ม
D1	CUSTOMER	Input
D11	TRANSACTION	Update
D12	CASH_01_mmdd	Update
D17	RECEIVE_TYPE	Input

รูปที่ 4.11 แสดงแผนภาพกระแสข้อมูล และที่เก็บข้อมูลของโปรแกรมประยุกต์ ส่วนที่ 6



Process	Frequency	Type	Program
P10	ตลอดวัน	Interactive Online	BATCH_PAID
P11	ตลอดวัน	Batch Online	REP_BATCH_PAID
P12	ตลอดวัน	Batch Online	UPDATE_PAID

Process	Frequency	Type	Program
P4	สิ้นเดือน,สิ้นปี	Batch Online	SUM_PAID
P4	สิ้นเดือน,สิ้นปี	Batch Online	SUM_NONPAID
P4	สิ้นเดือน,สิ้นปี	Batch Online	NON_PAID

Data Store	ชื่อแฟ้ม	การเข้าใช้แฟ้ม
D1	CUSTOMER	Input
D11	TRANSACTION	Update
D12	CASH_01_mmdd	Update
D17	RECEIVE_TYPE	Input

Data Store	ชื่อแฟ้ม	การเข้าใช้แฟ้ม
D1	CUSTOMER	Input
D11	TRANSACTION	Input
D36	PAID	Input

รูปที่ 4.12 แสดงแผนภาพกระแสข้อมูล และที่เก็บข้อมูลของโปรแกรมประยุกต์ ส่วนที่ 7

เพิ่มหลักแต่ละกลุ่มงาน						
ระบบทะเบียน เครื่องวิทยุ	ระบบเปิด-ปิด บริการลูกค้า	ระบบเปลี่ยน แปลงการใช้ บริการ	ระบบคิดค่าใช้จ่าย บริการ	ระบบจัดเตรียม และพิมพ์ใบแจ้ง หนี้	ระบบรับชำระ ค่าบริการ	ระบบบัญชี ส่วนกลาง, ภูมิภาค
CUSTOMER	CUSTOMER	CUSTOMER	CUSTOMER	CUSTOMER	CUSTOMER	CUSTOMER
MOBILE	MOBILE	MOBILE	CSCLT	TRANSACTION	TRANSACTION	TRANSACTION
MOB_CONTROL	MOB_HISTORY	MOB_HISTORY	EMX_TABLE	EMX_TABLE	CASH	CASH
MOB_MADE	MOB_CONTROL	MOB_CONTROL	MOB_CONTROL	TRUNK_TABLE	RECEIVE_TYPE	PAID_TYPE
MODEL	SERVICE_CLASS	SERVICE_CLASS	TRUNK_TABLE	CALL_CONTROL	BANK	BANK
	POSTCODE	POSTCODE	LOC_SWITCH, TIME, ZONE	BILL_DETAIL	CREDIT_BANK	CREDIT_BANK
	TEMP_CUST	TEMP_CUST	DOM_SWITCH, TIME, ZONE	PAID	GOVERNMENT	GOVERNMENT
	CALL_CONTROL	CALL_CONTROL	LOC_CALL, DOM_CALL, INT_CALL	LOC_CALL, DOM_CALL, INT_CALL	CLASS_GRP	CLASS_GRP
	BILL_DETAIL	BILL_DETAIL	CALL_CONTROL			PAID
		CUS_HISTORY	INTER_TABLE, TIME			
			SALE_PLAN_TBL			
			FREE_NUM			

ตารางที่ 4.2 แสดงเพิ่มหลักของแต่ละกลุ่มงาน

4.2.1.6. การจัดแบ่งฟังก์ชันการทำงาน

การจัดแบ่งฟังก์ชันการทำงาน แบ่งออกเป็น 3 กลุ่ม โดยพิจารณาถึงการอ้างอิงเพิ่มของฟังก์ชันข้ามกลุ่มในน้อยที่สุด จะมีข้อขัดข้องในการพิจารณาการกำหนดที่ตั้งเพิ่มกรณีดังรูปที่

1. หลังจากแบ่งกลุ่มแล้ว จะมีเพิ่มที่เป็นเอาพุท (Output) ของกลุ่มหนึ่ง และอีกกลุ่มจะทำการปรับปรุง(Update) เพิ่มนั้นเช่นเพิ่มชื่อ BILL_DETAIL.DAT, CALL_CONTROL.DAT, TRANSACTION.DAT และ PAID.DAT

2. หลังจากแบ่งกลุ่มแล้ว จะมีเพิ่มที่เป็นอินพุท (Input) กลุ่มหนึ่ง และอีกกลุ่มจะทำการปรับปรุง (Update) เพิ่มนั้น เช่น เพิ่มชื่อ CUSTOMER.DAT และ MOB_CONTROL.DAT

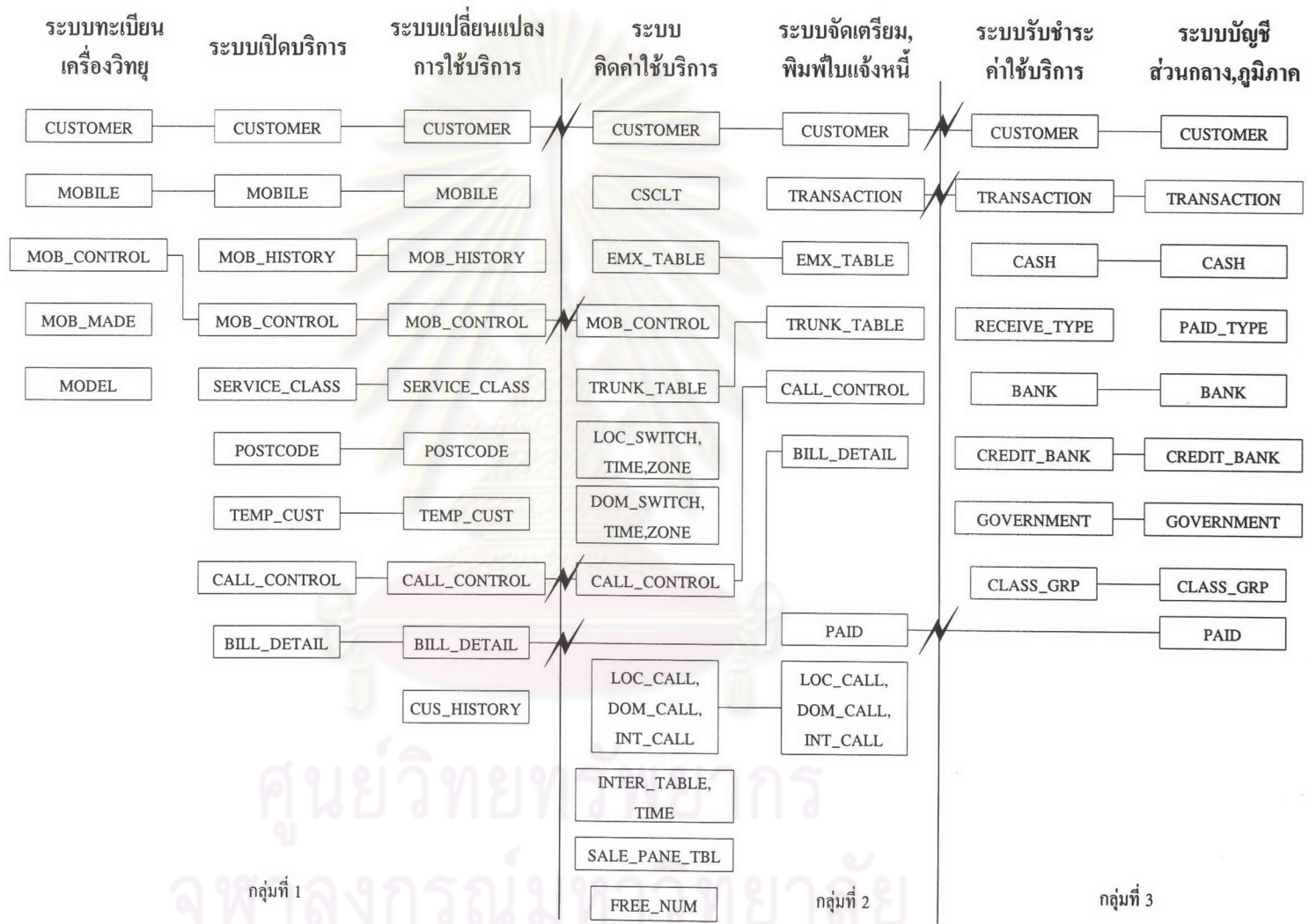
จะพิจารณากลุ่มเพิ่มที่เรียกใช้แบบปรับปรุง (Update) ด้วยเหตุผลต่อไปนี้

1. กลุ่มที่เรียกใช้เพิ่มเพื่อปรับปรุงเพิ่มจะมีความถี่ในการเรียกใช้งานบ่อยกว่าการเรียกใช้เพิ่มแบบเอาพุท

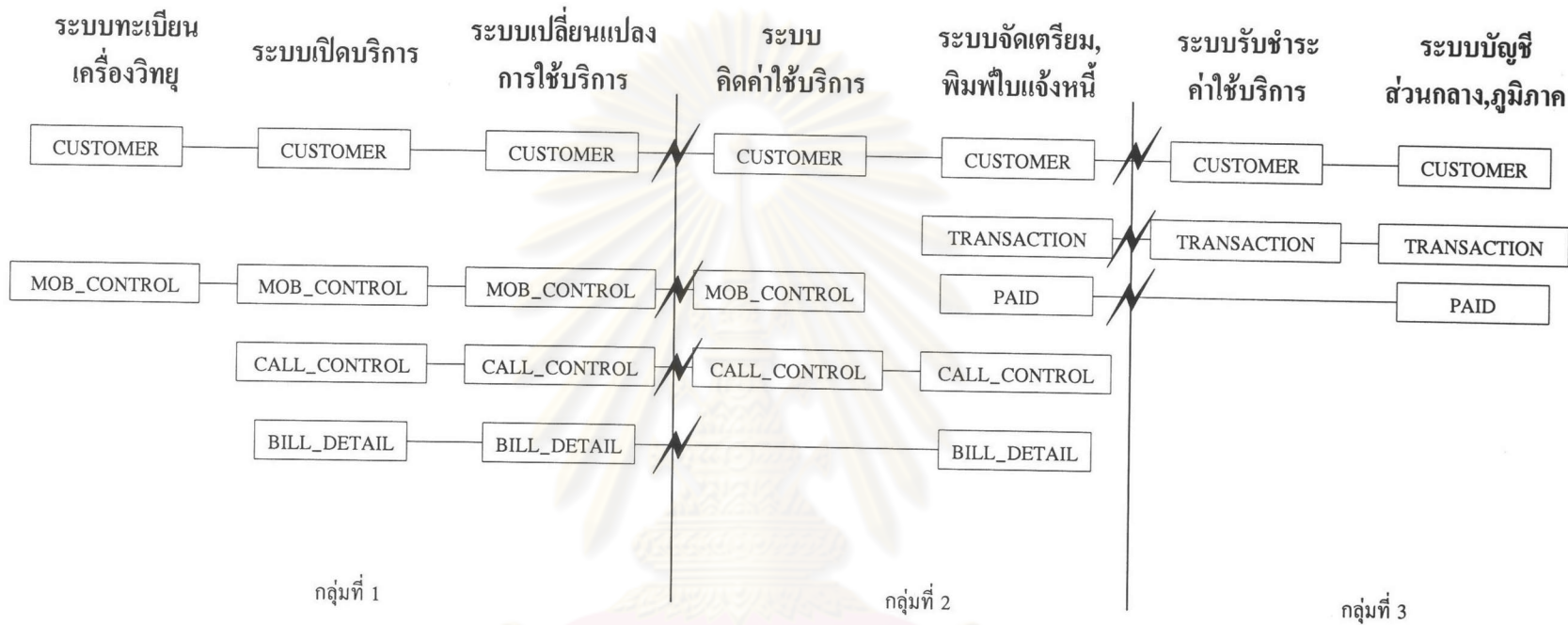
2. ถ้าเครือข่ายไม่สามารถเชื่อมต่อกันได้ ระหว่างบัพด้วยกรณีใดๆ กลุ่มที่เรียกใช้ แฟ้มแบบเอาพุท (Output) สามารถบันทึกแฟ้มไว้ที่บัพเฉพาะที่ (local Node) ก่อน เมื่อเครือข่าย ติดต่อกันได้แล้วสามารถถ่ายโอนแฟ้มไปยังแฟ้มที่บัพระยะไกล (remote node) ที่เป็นที่ตั้งแฟ้มจริง ได้ โดยเดคเนต และอาร์เอมเอสจะเป็นผู้ตรวจสอบเองว่า ถ้าเปิดแฟ้มที่กำหนดไว้ไม่ได้จะมีตัว เลือกที่ 2 ให้ไปแฟ้มที่ได้อีก



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.13 แสดงการแบ่งกลุ่มเพิ่มโดยพิจารณาจากการอ้างอิงเพิ่มข้ามกลุ่มน้อยที่สุด



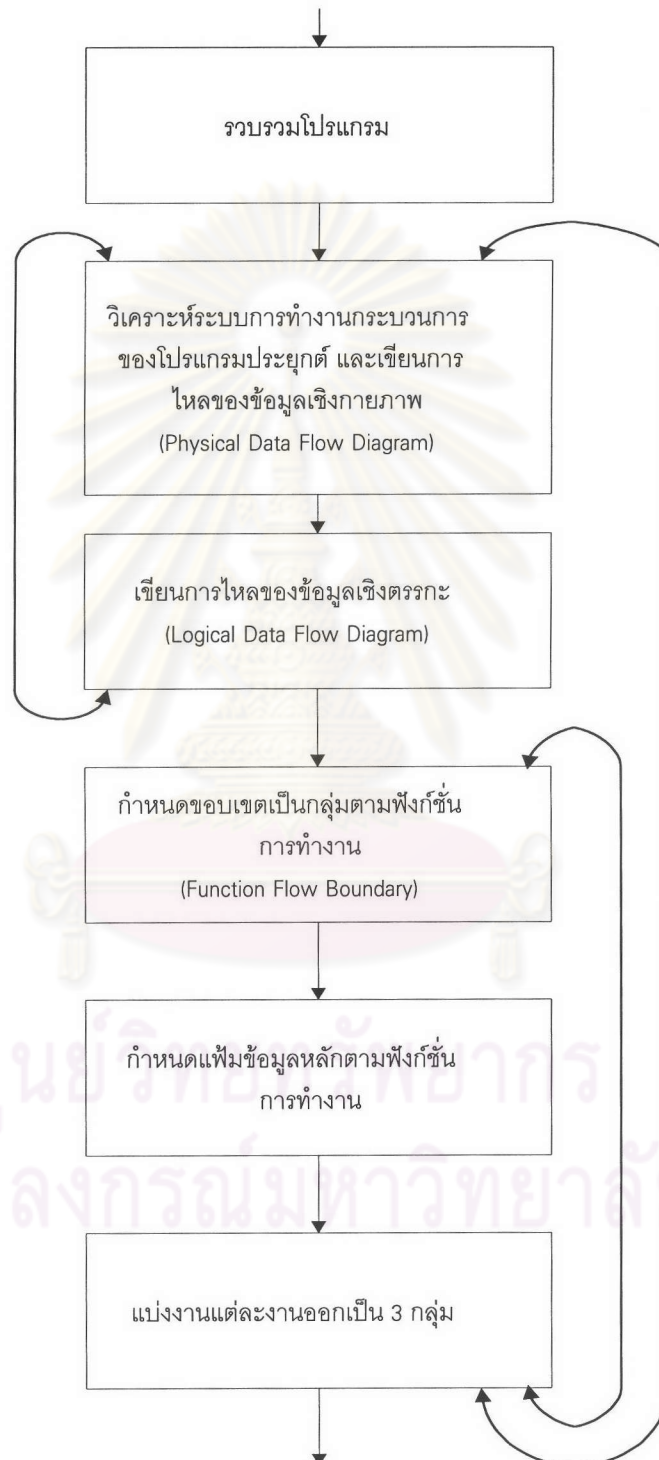
ชื่อแฟ้มที่มีการอ้างอิงข้ามกลุ่ม	การเข้าใช้แฟ้ม		
	กลุ่ม 1	กลุ่ม 2	กลุ่ม 3
1. CUSTOMER	Update	Input	Input
2. MOB_CONTROL	Update	Input	-
3. CALL_CONTROL	Output	Update	-
4. BILL_DETAIL	Output	Update	-
5. TRANSACTION	-	Output	Update
6. PAID	-	Output	Update

รูปที่ 4.14 แสดงการพิจารณาติดตั้งแฟ้มที่อ้างอิงข้ามกลุ่มไว้ที่กลุ่มใด ๆ ขึ้นอยู่กับระบบเรียกใช้แฟ้มนั้นแบบใด



รูปที่ 4.15 แสดงกลุ่มแฟ้มที่พิจารณากำหนดที่ตั้งแล้ว

ในการวิเคราะห์แต่ละขั้นตอน จะมีการทบทวนเพื่อให้แน่ใจว่าผลที่ได้จากการจัดกลุ่มโปรแกรมและแฟ้มได้ดีที่สุด (optimized)



รูปที่ 4.16 ขั้นตอนการวิเคราะห์ และกำหนดขอบเขตการทำงานของโปรแกรมประยุกต์

4.3 การกำหนดรูปแบบเครือข่าย (Network Configuring)

เมื่อสร้างเครือข่ายเดคเน็ต (DECnet) ให้เป็นระบบในองค์กร ต้องกำหนดรูปแบบ (configuration database) ของเครือข่าย เป็นการบันทึกรายละเอียดส่วนประกอบลงในระบบปฏิบัติการวีเอมเอส ประกอบด้วยคุณลักษณะของ บัพ (node), วงจรสื่อสาร (circuit), สายสื่อสาร (line), เส้นทางสื่อสาร (routing) โดยใช้โปรแกรมการควบคุมการเข้าถึงเครือข่าย (network access control)

การใช้โปรแกรมควบคุมเครือข่ายเอ็นซีพี (Network Control Program หรือ NCP) ของเดคเน็ต เพื่อเข้าไปทำการกำหนดรูปแบบฐานข้อมูล (configuration database) พารามิเตอร์เฉพาะต่างๆ ของส่วนประกอบเครือข่ายแต่ละตัวที่จะใช้ในระบบดังนี้

4.3.1 บัพ (Node)

บัพที่ใช้ซอฟต์แวร์เดคเน็ต (DECnet) บนระบบปฏิบัติการวีเอมเอส จะต้องกำหนดค่าที่ใช้เป็นพารามิเตอร์เพื่ออ้างอิงต่องานใช้งานดังนี้

4.3.1.1 ชื่อ และเลขที่อยู่ของบัพเดคเน็ต (DECnet Node Address and Name)

เพื่อกำหนดรูปแบบ (configure) การปฏิบัติการของเครือข่ายที่บัพเฉพาะที่สร้างฐานข้อมูลใช้กำหนดรูปแบบ (configuration database) สำหรับบัพเฉพาะที่ และบัพที่อยู่ติดกัน (adjacent node) นั่นคือเป็นการต่อวงจร (circuit) กำหนดชื่อ (name) และเลขที่อยู่ (address) ให้กับบัพทุกจุด ในเครือข่าย หลังจากที่ทำกรนี้แล้ว สามารถเข้าสู่บัพใดๆ ได้โดยชื่อของบัพที่กำหนดไว้แล้วนั้น

เพื่อให้ได้เส้นทาง (Routing) ตามต้องการ แต่ละบัพในเครือข่ายจะต้องมีเลขที่อยู่ (Address) เป็นหนึ่งเดียว ซึ่งเลขที่อยู่ของบัพนั้นมีรูปแบบเป็นตัวเลขดังนี้

เลขที่พื้นที่ . เลขที่บัพ (area-number.node-number)

- เมื่อ
1. เลขที่พื้นที่ (area-number) เป็นตัวเลขประจำพื้นที่ที่ บัพตั้งอยู่
 2. เลขที่บัพ (node-number) เป็นเลขที่อยู่ ของบัพภายในพื้นที่นั้น

เลขที่พื้นที่ (area-number) แต่ละหมายเลขในเครือข่ายจะต้องไม่ซ้ำกัน และเลขที่บัพ (node-number) แต่ละหมายเลขภายในพื้นที่ (area) จะต้องไม่ซ้ำกันด้วย

ลักษณะของเลขประจำตัวบัพ (node identification) มี 2 รูปแบบ คือ เลขที่อยู่ของบัพ (node address) และชื่อบัพ (node name) ซึ่งเลขที่อยู่ของบัพ (node address) นั้นใช้รายละเอียดเป็นตัวเลขดังได้กล่าวมาแล้ว กำหนดแต่ละบัพอยู่ในฐานข้อมูลใช้กำหนดรูปแบบ (configuration

database) ส่วนชื่อบัพ (node name) นั้น สามารถใช้ตัวอักษรได้เป็นการง่ายต่อการจำชื่อบัพ (Node Name) มากกว่าการจำเลขที่อยู่ (Node address) และอาจจำทั้งชื่อ และเลขที่อยู่ไปใช้งานก็ได้

4.3.1.2 การกำหนดชื่อบัพ และเลขที่อยู่

เพื่อกำหนดรูปแบบบัพในเครือข่ายเดคเน็ต เดคเน็ตมีโปรแกรมควบคุมเครือข่าย (Network Control Program หรือ NCP) ช่วยในการกำหนดรูปแบบ (configure)

ขั้นแรกต้องกำหนดรูปแบบของบัพทั้งหมด ในเครือข่ายเดคเน็ตทั้งสามบัพ สามารถใช้คำสั่ง “NETCONFIG.COM” ภายใต้สารบบ (Directory) SYS\$MANAGER

ในที่ตั้งชื่อบัพ ดังนี้

บัพแรกชื่อ	“CATBUS”	เลขที่พื้นที่ 50	เลขที่อยู่ 1
บัพที่สองชื่อ	“CATRAD”	เลขที่พื้นที่ 9	เลขที่อยู่ 1
บัพที่สามชื่อ	“CATACT”	เลขที่พื้นที่ 30	เลขที่อยู่ 1

4.3.2 วงจรสื่อสาร (Circuits)

เป็นเส้นทางการติดต่อสื่อสารระดับสูงระหว่างบัพ เกิดขึ้นเหนือวงจร และวงจรมันปฏิบัติการเหนือสายสื่อสารทางกายภาพ (physical line) ซึ่งเป็นเส้นทางการติดต่อสื่อสารระดับล่างสุด วงจรอีเทอเน็ตแลน (Ethernet LAN circuit) จัดให้สำหรับการติดต่อแบบเข้าถึงหลายวงจร (multi-access) ระหว่างบัพ หลายบัพที่อยู่บนสื่อกลางที่การติดต่อจะแพร่กระจายไปตลอดบนสายสื่อกลาง (broadcast medium) ทุกๆ บัพต้องมีเลขประจำตัว เป็นหนึ่งเดียวภายในพื้นที่ของบัพนั้น

อุปกรณ์วงจรอีเทอเน็ต (Ethernet Circuit Devices)

อุปกรณ์ของวงจรอีเทอเน็ตทางกายภาพ ที่ใช้ในการวิจัยครั้งนี้ชื่อ ดีควินา (DEQNA) อุปกรณ์นี้เหมือนกับ Ethernet/802.3 protocol มีซอฟต์แวร์ไดรเวอร์ (Software Driver) ชื่อ เอกซ์คิว (XQ) และอุปกรณ์ทางกายภาพ และซอฟต์แวร์ไดรเวอร์ จะเป็นตัวกำหนดชื่อวงจรในที่นี้ใช้ชื่อ วงจรคิวเอนเอ (QNA)

อุปกรณ์นี้ใช้เป็นวงจรอีเทอร์เน็ต ข่าวสารข้อความอีเทอเน็ต ที่ส่งออกไปเป็นเดตาแกรม (datagram) นั้น สามารถสูญหายได้ อาจเกิดจากการส่งผิดพลาดและอื่นๆ เดคเน็ต (DECnet) จัดให้ข่าวสารข้อความที่สูญเสียนำใหม่ (retransmission) โดยอัตโนมัติ บนวงจรอีเทอเน็ต (Ethernet) และอุปกรณ์ไดรเวอร์นี้สามารถใช้พิธีการ (protocol) ได้หลายชนิดพร้อมกัน คือ ขณะที่

เดคเน็ต (DECnet) ทำงานอยู่บนโปรแกรมประยุกต์ (application) สามารถใช้วิธีการชนิดอื่นบนอุปกรณ์อีเทอร์เน็ตเดียวกันได้

4.3.3 สายสัญญาณ (Lines)

สายสัญญาณ (lines) จัดเป็นวัตถุสื่อสาร และเป็นเส้นทางสื่อสารในระดับล่างสุด โดยวงจรเป็นเส้นทางสื่อสารระดับสูง ปฏิบัติการเหนือสายสัญญาณ (line) อีกทีหนึ่ง

ชนิดของสายสัญญาณที่ใช้บนเดคเน็ต (DECnet for OpenVMS Lines)

เดคเน็ตบนระบบปฏิบัติการวีเอ็มเอส (DECnet for OpenVMS) สนับสนุนสายสัญญาณ (line) ได้ 4 ชนิดคือ ดีดีซีเอ็มพี (DDCMP), ซีไอ (Computer Interconnect หรือ CI), อีเทอร์เน็ต (Ethernet) และเอฟดีดีไอ (FDDI)

ในที่นี้ใช้สายสัญญาณเฉพาะชนิดของการต่ออีเทอร์เน็ต ซึ่งเป็นการต่อแบบนี้สามารถเข้าถึงได้หลายบัพพร้อมกัน (multi-access) บนสายสัญญาณเดียว

4.3.4 เส้นทางสื่อสาร (Routing)

เดคเน็ตสามารถกำหนดบัพให้เป็นแบบบัพเส้นทางผ่าน (routing node) และบัพไม่ใช่เส้นทางผ่าน หรือบัพตอนปลาย (non-routing หรือ end node)

บัพเส้นทางผ่าน (routing) มีหน้าที่กำหนดทางเดินหรือเส้นทางซึ่งข้อมูล (packet) เดินทางไปสู่ปลายทาง ในชั้นเส้นทาง (routing layer) ของเดคเน็ต (DECnet) ควบคุมการกำหนดเส้นทาง (routing function) เพราะว่าการกระจายเส้นทางในเครือข่าย จะเป็นไปตามที่กระทำไว้ตอนกำหนดรูปแบบ (configuration) ของเครือข่ายไว้

บัพไม่ใช่เส้นทางผ่าน หรือบัพตอนปลาย (non-routing หรือ end node) เป็นบัพที่ไม่สามารถใช้ติดต่อกับเครือข่ายได้ ถ้าจะติดต่อไปเครือข่ายอื่นต้องผ่านบัพเส้นทาง

ในที่นี้การวิจัยกำหนดการเชื่อมต่อแบบอีเทอร์เน็ตแลนบัพทุกตัว สามารถติดต่อกันถึงกันโดยไม่ต้องผ่านตัวกลาง ดังนั้นทุกบัพจะถูกกำหนดเป็นบัพปลายทางทั้งหมด

4.3.5 การค้นหาต้นเหตุที่กระบวนการข้ามเครือข่ายผิดพลาด

ถ้าเกิดสถานะที่การเชื่อมต่อเกิดผิดพลาดขึ้น ระหว่างบัพสามารถเข้าไปดูแฟ้มลงบันทึก (log file) ชื่อ NETSERVER.LOG ที่แฟ้มระยะไกลได้ ซึ่งมีหลายสาเหตุ เช่น

1. ไม่สามารถเข้าสู่กระบวนการลงบันทึกเข้า (login) หรือเข้าใช้แฟ้มใดๆ
2. ไม่ผ่านระบบป้องกันการเข้าไปดำเนินงานกับโปรแกรมบนเครือข่าย

3. การลงบันทึกเข้าทุกครั้ง จะมีการสั่งให้ไปดำเนินการกับแฟ้มชุดคำสั่งได้โดยอัตโนมัติเพื่อเข้าไปวิ่งโปรแกรมบางอย่างให้กับผู้ใช้กรณีเป็นแบบวิธีโต้ตอบ (interactive mode) แต่เนื่องจากการเข้าถึงแฟ้มระยะไกล ก็ยังต้องผ่านกระบวนการนี้ เพื่อป้องกันการเกิดข้อขัดข้องขึ้น สามารถใส่ชุดคำสั่งดีซีแอล (DCL) ไว้ในแฟ้มชุดคำสั่งชื่อ LOGIN.COM หรือตามที่กำหนดไว้แตกต่างจากนี้ในการอนุญาตเข้าใช้ระบบของแต่ละผู้ใช้ด้วยคำสั่งนี้ เพื่อให้ข้ามบางคำสั่งที่ใช้เฉพาะแบบโต้ตอบหรือกลุ่ม

```
$ IF F$MODE().EQS."NETWORK" THEN
```

หมายเหตุ

1. \$ เป็นข้อความพร้อมรับดีซีแอล (DCL Prompt)
2. IF THEN เป็นคำสั่งดีซีแอล ใช้ตรวจสอบเพื่อการตัดสินใจ
3. F\$MODE() เป็นคำสั่งดีซีแอล ใช้ตรวจสอบสถานะขณะนั้นว่าเป็นแบบใด
4. .EQS. เป็นคำสั่งดีซีแอลบูลีน ใช้ตรวจสอบการเท่ากับของข้อความ (Equal String)
5. "NETWORK" คำอักขระที่ใช้เปรียบเทียบ

4.4 การควบคุมการผ่านเข้ามาใช้บัพตัวแทน (Proxy Login Access Control)

การลงบันทึกเข้าโดยผ่านบัพตัวแทน (Proxy login) เพื่อให้ผู้ใช้ที่บัพระยะไกล (remote node) เข้าไปใช้ผ่านบัพเฉพาะที่ (local node) โดยไม่ต้องมีการส่งข่าวสารการควบคุมการเข้าถึง (access control information) คือไม่ต้องมีชื่อผู้ใช้ (User Name) และรหัสผ่าน (Password) สามารถทำงานข้ามบัพได้ทันที การเข้ามาโดยผ่านบัพตัวแทน (Proxy login) ไม่เหมือนกับการเข้ามาใช้งานแบบโต้ตอบ (interactive login) ต้องการให้ผู้ใช้ป้อนชื่อผู้ใช้และรหัสผ่านก่อน ที่ผู้ใช้สามารถกระทำ การปฏิบัติแบบโต้ตอบ (interactive operation) ได้ ใดๆ โดยบนบัพระยะไกลจะใช้ค่าเอกสิทธิ์ (privilege) ที่กำหนดอยู่บนบัพที่ผู้ใช้ลงบัญชีไว้เป็นค่าโดยปริยาย (default)

ในการกำหนดการลงบันทึกผ่านตัวแทน (Proxy Login) ต้องลงบัญชี (account) ที่บัพเฉพาะที่ และลงบัญชีตัวแทน (proxy) ที่บัพระยะไกลด้วย ทุกๆ บัพที่ต้องการเข้าถึงแฟ้มระยะไกล

4.4.1 การลงบัญชีตัวแทน (Proxy Account)

การลงบัญชีตัวแทน (proxy account) นั้น อนุญาตให้ผู้ใช้บนบัพระยะไกล (remote node) เข้าใช้แฟ้มได้โดยไม่ต้องมีบัญชีส่วนตัว (private account) ผู้ใช้ระยะไกลสามารถใช้คำสั่งเพื่อเข้าถึงข้อมูลได้ทุกบัพ ที่ผู้ใช้นั้นมีการลงบัญชีเข้าถึงแบบตัวแทน (proxy access)

การควบคุมการใช้โดยการลงบัญชีตัวแทน ที่บัพเฉพาะที่ใช้โปรแกรมอรรถประโยชน์อินเทอร์เน็ต (Authorize Utility) ไปเพิ่ม และแก้ไขในแฟ้มข้อมูลพ็อกซี่ (proxy) คือแฟ้ม NETPROXY.DAT ชื่อผู้ใช้ และชื่อบัพบนบัพระยะไกลจะต้องเหมือนกันกับที่บัพเฉพาะที่

4.4.2 กำหนดกลุ่มงานที่ได้แบ่งไว้ลงบนบัพทั้งสาม

ตามที่ได้กำหนดบัพที่ใช้ในการวิจัย 3 บัพ ให้มีชื่อ CATBUS, CATRAD และ CATACT ตามลำดับไว้แล้วพิจารณานำแฟ้มที่เป็นเอนทิตีพื้นฐานของแต่ละระบบงานทั้งสามกลุ่มติดตั้งลงบนแต่ละบัพทั้ง 3 ตามตารางที่ 4.3

บัพชื่อ "CATBUS"	บัพชื่อ "CATRAD"	บัพชื่อ "CATACT"
1. ระบบทะเบียนเครื่องวิทยุ	1. ระบบคิดค่าใช้บริการ	1. ระบบรับชำระค่าใช้บริการ
2. ระบบเปิด-ปิดบริการลูกค้า	2. ระบบจัดเตรียมและพิมพ์ใบแจ้งหนี้	2. ระบบบัญชีส่วนกลาง ภูมิภาค
3. ระบบเปลี่ยนแปลงการใช้บริการ		

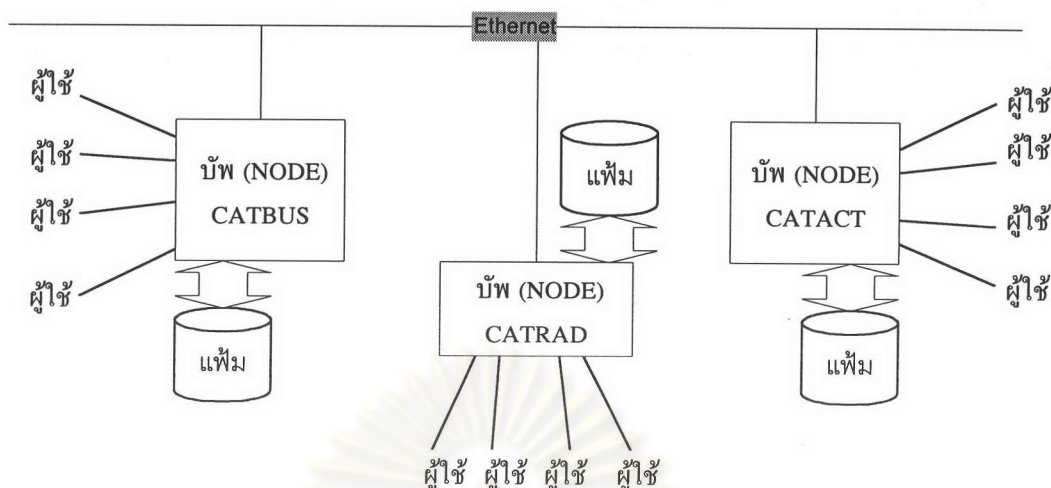
ตารางที่ 4.3 แสดงการกำหนดการกระจายฟังก์ชันการทำงานของโปรแกรมประยุกต์ลงบนบัพทั้ง 3

โดยทั้งสามกลุ่มงานจากตารางที่ 4.4 แต่ละกลุ่มประกอบด้วย

1. แฟ้ม (Files)
2. ผู้ใช้ (Users)
3. โปรแกรมประยุกต์ (Applications)

แฟ้มข้อมูลที่โปรแกรมประยุกต์ใช้ดำเนินงานทั้งหมด จะถูกแบ่งเป็นกลุ่ม 3 กลุ่มตามรูปที่ 4.17จะนำไปติดตั้งตามบัพที่กำหนด 3 บัพให้ตรงตามบัพที่ถูกกำหนดหน้าที่ไว้แล้ว

ส่วนจอเทอร์มินัลที่เป็นอุปกรณ์รับเข้า-ส่งออกสารสนเทศ สำหรับให้ผู้ใช้ปฏิบัติงานจะต้องระบุว่าใช้ทำงานในฟังก์ชันใด เนื่องจากโปรแกรมประยุกต์จะติดตั้งไว้ทั้ง 3 บัพ เพื่อให้ผู้ใช้สามารถปฏิบัติงานได้ที่เทอร์มินัลทุกเครื่อง โดยมองผ่านแฟ้มข้อมูลเหมือนกันหมด แต่ถ้าผู้ใช้ปฏิบัติงานนอกเหนือจากฟังก์ชันที่ระบุไว้แต่ละเทอร์มินัล จะมีผลด้านความเร็วเนื่องจากการเข้าถึงแฟ้มหลักของการทำงาน ผู้ใช้คนนั้นไม่ได้ติดตั้งไว้ที่บัพนั้น จะทำให้เกิดการเข้าถึงแฟ้มข้ามเครือข่ายสูงมาก



รูปที่ 4.17 รูปแบบของเครือข่ายคอมพิวเตอร์ (Computer Network Configuration) ที่ได้ออกแบบ

เมื่อได้ติดตั้งระบบงานแต่ละบัพแล้ว ผู้ใช้ที่มีหน้าที่เกี่ยวข้องในการปฏิบัติงานในแต่ละระบบงานจะต้องดำเนินงานโปรแกรมประยุกต์ เฉพาะบัพที่ติดตั้งระบบงาน เนื่องจากแฟ้มข้อมูลที่ได้แบ่งกลุ่มไว้เป็นเอนทิตีพื้นฐานของหน่วยงานที่ผู้ใช้นั้นสังกัดอยู่นั้นก็คือ ความบ่อยในการเรียกใช้งาน การปรับปรุงแฟ้มจะดำเนินงานที่บัพเฉพาะที่เป็นส่วนมาก จะมีการเข้าถึงแฟ้มระยะไกลบ้างก็เฉพาะต้องการอ่านข้อมูลข้ามหน่วยงานบ้างเท่านั้น การติดตั้งระบบงานดำเนินงานตามขั้นตอน ดังนี้

1. ทำการเชื่อมต่อเครื่องเทอร์มินัลทางกายภาพกับบัพ ให้ตรงกับที่แฟ้มข้อมูลระบบงานติดตั้งอยู่ โดยระบุว่าเครื่องเทอร์มินัลใดทำงานอะไร สำหรับหน่วยงานใด

บัพชื่อ "CATBUS"	บัพชื่อ "CATRAD"	บัพชื่อ "CATACT"
1. ระบบทะเบียนเครื่องวิทยุ	1. ระบบคิดค่าใช้บริการ	1. ระบบรับชำระค่าใช้บริการ
2. ระบบเปิด-ปิดบริการลูกค้า	2. ระบบจัดเตรียมและพิมพ์ใบแจ้งหนี้	2. ระบบบัญชีส่วนกลางภูมิภาค
3. ระบบเปลี่ยนแปลงการใช้บริการ		

ตารางที่ 4.4 แสดงระบบงานประยุกต์ติดตั้งตามบัพที่กำหนด จากที่การออกแบบไว้

2. บันทึกลงบัญชี (account) ผู้ใช้แต่ละบุคคลบนบัพที่แฟ้มระบบงานติดตั้งอยู่ (local node) โดยใช้คำสั่งอรรถประโยชน์จัดการระบบชื่อ AUTHORIZE เป็นเครื่องมือการจัดการระบบใช้

ควบคุมการเข้าใช้ระบบ โดยการกำหนดชื่อบัญชีเพื่อให้สำหรับคำนวณเปรียบเทียบประสิทธิภาพการเรียกใช้ระบบก่อน และหลังการติดตั้ง มีรูปแบบคำสั่งดังนี้

\$ RUN AUTHORIZE

UAF> ADD <newusername>/ACCOUNT=account-name/PASSWORD=(password)

1. \$ เป็นข้อความพร้อมรับดีซีแอล (DCL Prompt)
2. RUN AUTHORIZE เป็นคำสั่งดีซีแอล ใช้ดำเนินงานโปรแกรม Authorize
3. UAF เป็นข้อความพร้อมรับยูเอเอฟ (UAF Prompt)
4. ADD เป็นคำสั่งเพิ่มผู้ใช้ โดยกำหนดชื่อผู้ใช้เป็นพารามิเตอร์
5. /ACCOUNT ตามด้วยชื่อบัญชีเป็นพารามิเตอร์
6. /PASSWORD ตามด้วยรหัสผ่านของผู้ใช้

บัญชีชื่อ "CATBUS"	บัญชีชื่อ "CATRAD"	บัญชีชื่อ "CATACT"
ชื่อผู้ใช้ INFnnn /ชื่อบัญชี REGISTER	ชื่อผู้ใช้ ENGnnn /ชื่อบัญชี ENGINEER	ชื่อผู้ใช้ FINnnn /ชื่อบัญชี FINANCE
ชื่อผู้ใช้ BUSnnn /ชื่อบัญชี BUSINESS	ชื่อผู้ใช้ EMXnnn /ชื่อบัญชี EMX	ชื่อผู้ใช้ ACTnnn /ชื่อบัญชี ACCOUNT
	ชื่อผู้ใช้ INF005,ENG045 /ชื่อบัญชี INSPECT	

ตารางที่ 4.5 แสดงกลุ่มผู้ใช้ที่กำหนดให้ปฏิบัติงานบัญชีออกแบบไว้

3. ลงบันทึกบัญชีตัวแทน (proxy account) ให้ผู้ใช้บนบัพระยะไกลที่เหลือ เพื่อให้ผู้ใช้สามารถเข้าถึงแฟ้มข้อมูลข้ามระบบงานได้ โดยใช้คำสั่งอรรถประโยชน์จัดการระบบชื่อ AUTHORIZE.EXE

\$ RUN AUTHORIZE.EXE

UAF> ADD/PROXY node::remote-user local-user

คำอธิบายความหมายของคำสั่ง

1. \$ เป็นข้อความพร้อมรับดีซีแอล (DCL Prompt)
2. RUN AUTHORIZE เป็นคำสั่งดีซีแอล ใช้ดำเนินงานโปรแกรม Authorize
3. UAF เป็นข้อความพร้อมรับยูเอเอฟ (UAF Prompt)

4. ADD/PROXY เป็นคำสั่งเพิ่มผู้ใช้ ที่เป็นตัวแทนเข้าใช้ระบบข้ามเครือข่าย
5. node เป็นชื่อบัพระยะไกลที่ผู้ใช้มีบัญชีอยู่
6. remote-user ชื่อผู้ใช้ระยะไกลที่ต้องการข้ามมาใช้เพิ่ม
7. local-user ชื่อผู้ใช้เฉพาะที่

บัพชื่อ "CATBUS"	บัพชื่อ "CATRAD"	บัพชื่อ "CATACT"
CATRAD::ENGnnn ENGnnn	CATACT::FINnnn FINnnn	CATRAD::ENGnnn ENGnnn
CATRAD::EMXnnn EMXnnn	CATACT::ACTnnn ACTnnn	CATRAD::EMXnnn EMXnnn
CATRAD::INF005 INF005		CATRAD::INF005 INF005
CATRAD::ENG045 ENG045		CATRAD::ENG045 ENG045
CATACT::FINnnn FINnnn	CATBUS::INFnnn INFnnn	CATBUS::INFnnn INFnnn
CATACT::ACTnnn ACTnnn	CATBUS::BUSnnn BUSnnn	CATBUS::BUSnnn BUSnnn

ตารางที่ 4.6 แสดงการใช้คำสั่งเปิดบัญชีตัวแทน เพื่อการเข้าถึงแฟ้มระยะไกล

โดยไม่ต้องใช้รหัสผ่าน

4. ทำการสร้างแฟ้มเก็บชุดคำสั่ง ที่ใช้กำหนดแฟ้มทางตรรกะ ให้กับโปรแกรมประยุกต์ ตามตำแหน่งที่แฟมนั้นติดตั้งอยู่ เพื่อให้ผู้ใช้ดำเนินงานกับโปรแกรมประยุกต์ได้โดยที่ไม่ต้องรู้ที่ตั้งแฟ้มถูกกระจายไปตามบัพต่างๆ แล้ว โดยใช้คำสั่งอรรถประโยชน์จัดการระบบกำหนดชื่อแฟ้มเชิงตรรกะ

\$ ASSIGN/SYSTEM node::device:[directory.subdirectory]filename.type;version logicalfilename

คำอธิบายความหมายของคำสั่ง

1. \$ เป็นข้อความพร้อมรับดีซีแอล (DCL Prompt)
2. ASSIGN เป็นคำสั่งดีซีแอล ใช้กำหนดชื่อแฟ้มเชิงตรรกะ
3. /SYSTEM = ชื่อแฟ้มเชิงตรรกะนี้กำหนดให้ผู้ใช้ทุกกระบวนการในระบบใช้
4. node = ชื่อบัพที่แฟ้มติดตั้งอยู่
5. device = ชื่ออุปกรณ์ที่แฟ้มติดตั้ง
6. directory และ subdirectory ชื่อสารบบที่แฟ้มติดตั้งอยู่
7. filename = ชื่อแฟ้ม
8. type = ชนิดของแฟ้ม
9. version = รุ่นของแฟ้ม
10. logicalfilename = ชื่อแฟ้มเชิงตรรกะ

การกำหนดตารางอ้างอิงเพิ่มทางตรรกะ (Logical File Name) บนแต่ละบัพให้กับโปรแกรม
ประยุกต์

บนบัพชื่อ CATBUS

\$ ASSIGN/SYSTEM DUA0:[CAT.DAT]CUSTOMER.DAT CUSTOMER
 \$ ASSIGN/SYSTEM DUA0:[CAT.DAT]MOBILE.DAT MOBILE
 \$ ASSIGN/SYSTEM DUA0:[CAT.DAT]MOB_HISTORY.DAT MOB_HISTORY
 \$ ASSIGN/SYSTEM DUA0:[CAT.DAT]MOB_CONTROL.DAT MOB_CONTROL
 \$ ASSIGN/SYSTEM DUA0:[CAT.DAT]SERVICE_CLASS.DAT SERVICE_CLASS
 \$ ASSIGN/SYSTEM DUA0:[CAT.DAT]POSTCODE.DAT POSTCODE
 \$ ASSIGN/SYSTEM DUA0:[CAT.DAT]TEMP_CUST.DAT TEMP_CUST
 \$ ASSIGN/SYSTEM DUA0:[CAT.DAT]CUS_HISTORY.DAT CUS_HISTORY
 \$ ASSIGN/SYSTEM DUA0:[CAT.DAT]MOB_MADE.DAT MOB_MADE
 \$ ASSIGN/SYSTEM DUA0:[CAT.DAT]MODEL.DAT MODEL

 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]CSCLT.DAT CSCLT
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]EMX_TABLE.DAT EMX_TABLE
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]TRUNK_TABLE.DAT TRUNK_TABLE
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]LOC_SWITCH.DAT LOC_SWITCH
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]LOC_TIME.DAT LOC_TIME
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]LOC_ZONE.DAT LOC_ZONE
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]DOM_SWITCH.DAT DOM_SWITCH
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]DOM_TIME.DAT DOM_TIME
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]DOM_ZONE.DAT DOM_ZONE
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]INTER_TABLE.DAT INTER_TABLE
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]INTER_TIME.DAT INTER_TIME
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]CALL_CONTROL.DAT CALL_CONTROL
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]LOC_CALLnn.DAT LOC_CALLnn
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]DOM_CALLnn.DAT DOM_CALLnn
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]BILL_DETAIL.DAT BILL_DETAIL
 \$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]FREE_NUM.DAT FREE_NUM

 \$ ASSIGN/SYSTEM CATACT::DUA0:[CAT.DAT]TRANSACTION.DAT TRANSACTION
 \$ ASSIGN/SYSTEM CATACT::DUA0:[CAT.DAT]CASHnn.DAT CASHnn
 \$ ASSIGN/SYSTEM CATACT::DUA0:[CAT.DAT]RECEIVE_TYPE.DAT RECEIVE_TYPE
 \$ ASSIGN/SYSTEM CATACT::DUA0:[CAT.DAT]BANK.DAT BANK
 \$ ASSIGN/SYSTEM CATACT::DUA0:[CAT.DAT]CREDIT_BANK.DAT CREDIT_BANK
 \$ ASSIGN/SYSTEM CATACT::DUA0:[CAT.DAT]GOVERNMENT.DAT GOVERNMENT
 \$ ASSIGN/SYSTEM CATACT::DUA0:[CAT.DAT]CLASS_GRP.DAT CLASS_GRP
 \$ ASSIGN/SYSTEM CATACT::DUA0:[CAT.DAT]PAID.DAT PAID
 \$ ASSIGN/SYSTEM CATACT::DUA0:[CAT.DAT]PAID_TYPE.DAT PAID_TYPE

ตารางที่ 4.7 แสดงชุดคำสั่งใช้กับโปรแกรมประยุกต์ อ้างอิงที่ดั่งเพิ่มที่ออกแบบไว้

การกำหนดตารางการอ้างอิงเพิ่มทางตรรกะ (Logical File Name) บนแต่ละบัพให้กับโปรแกรม
ประยุกต์

บัพชื่อ CATACT

```

$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]CUSTOMER.DAT CUSTOMER
$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]MOBILE.DAT MOBILE
$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]MOB_HISTORY.DAT MOB_HISTORY
$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]MOB_CONTROL.DAT MOB_CONTROL
$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]SERVICE_CLASS.DAT SERVICE_CLASS
$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]POSTCODE.DAT POSTCODE
$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]TEMP_CUST.DAT TEMP_CUST
$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]CUS_HISTORY.DAT CUS_HISTORY
$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]MOB_MADE.DAT MOB_MADE
$ ASSIGN/SYSTEM CATBUS::DUA0:[CAT.DAT]MODEL.DAT MODEL

$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]CSCLT.DAT CSCLT
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]EMX_TABLE.DAT EMX_TABLE
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]TRUNK_TABLE.DAT TRUNK_TABLE
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]LOC_SWITCH.DAT LOC_SWITCH
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]LOC_TIME.DAT LOC_TIME
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]LOC_ZONE.DAT LOC_ZONE
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]DOM_SWITCH.DAT DOM_SWITCH
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]DOM_TIME.DAT DOM_TIME
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]DOM_ZONE.DAT DOM_ZONE
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]INTER_TABLE.DAT INTER_TABLE
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]INTER_TIME.DAT INTER_TIME
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]CALL_CONTROL.DAT CALL_CONTROL
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]LOC_CALLnn.DAT LOC_CALLnn
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]DOM_CALLnn.DAT DOM_CALLnn
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]BILL_DETAIL.DAT BILL_DETAIL
$ ASSIGN/SYSTEM CATRAD::DUA0:[CAT.DAT]FREE_NUM.DAT FREE_NUM

$ ASSIGN/SYSTEM DUA0:[CAT.DAT]TRANSACTION.DAT TRANSACTION
$ ASSIGN/SYSTEM DUA0:[CAT.DAT]CASHnn.DAT CASHnn
$ ASSIGN/SYSTEM DUA0:[CAT.DAT]RECEIVE_TYPE.DAT RECEIVE_TYPE
$ ASSIGN/SYSTEM DUA0:[CAT.DAT]BANK.DAT BANK
$ ASSIGN/SYSTEM DUA0:[CAT.DAT]CREDIT_BANK.DAT CREDIT_BANK
$ ASSIGN/SYSTEM DUA0:[CAT.DAT]GOVERNMENT.DAT GOVERNMENT
$ ASSIGN/SYSTEM DUA0:[CAT.DAT]CLASS_GRP.DAT CLASS_GRP
$ ASSIGN/SYSTEM DUA0:[CAT.DAT]PAID.DAT PAID
$ ASSIGN/SYSTEM DUA0:[CAT.DAT]PAID_TYPE.DAT PAID_TYPE

```

ตารางที่ 4.7 แสดงชุดคำสั่งใช้กับโปรแกรมประยุกต์ อ้างอิงที่ตั้งเพิ่มที่ออกแบบไว้ (ต่อ)

5. ติดตั้งโปรแกรมประยุกต์ลงบนทุกบัพ ให้เหมือนกัน
6. บันทึกชุดคำสั่ง F\$MODE().EQS.“NETWORK” ลงบนแฟ้ม “login.com” ของผู้ใช้

4.5 การทดสอบก่อนการติดตั้งระบบเพื่อทดสอบเวลาตอบสนอง

ระบบปฏิบัติการวีเอ็มเอส มีรันไทม์ไลบรารีรูทีน (Run-Time Library routines) ไว้ให้โปรแกรมเรียกใช้สำหรับวัดสมรรถนะของโปรแกรมประยุกต์ โดยมีรูทีนให้ดังนี้ LIB\$INIT_TIMER และ LIB\$SHOW_TIMER ใช้สำหรับเก็บเวลา และแสดงเวลาที่โปรแกรมนั้นเรียกใช้ทรัพยากร รูทีน LIB\$INIT_TIMER จะกำหนดที่เก็บค่าสถิติที่นับ และเก็บค่าไว้ที่ได้ของโปรแกรม ส่วนรูทีน LIB\$SHOW_TIMER แสดงค่าที่เก็บไว้ โดยรูทีน LIB\$INIT_TIMER แล้วเขียนออกไปยังเทอมนัลในรูปแบบรหัสแอสกี (ASCII code) เป็นข้อความที่ปรากฏดังนี้

ELAPSED: hh:mm:ss.99 CPU: h:mm:ss.99 BUFIO: n DIRIO: n FAULTS: nn

1. ELAPSED, เวลาที่ใช้ทั้งหมดของกระบวนการ แสดงรายละเอียด 1/100 วินาที
2. CPU, เวลาที่กระบวนการเข้าไปใช้ซีพียู แสดงรายละเอียด 1/100 วินาที
3. BUFIO จำนวนครั้งที่เรียกใช้ บัฟเฟอร์ไอโอ (Buffer Input/Output) หมายถึง จำนวนครั้งที่ใช้เทอมนัล หรือ เครื่องพิมพ์
4. DIRIO จำนวนครั้งที่ใช้ ไอโอโดยตรง (Direct Input/Output) หมายถึง จำนวนครั้งที่ใช้งานแม่เหล็ก หรือเทปแม่เหล็ก
5. FAULTS จำนวนครั้งที่เกิดหน้าบกพร่อง

โปรแกรมทดสอบที่สร้างขึ้นใช้กับตัวแปลภาษา โคบอล และเรียกรูทีน LIB\$INIT_TIMER และ LIB\$START_TIMER เพื่อใช้ทดสอบเวลาตอบสนองของโปรแกรมประยุกต์ของการเรียกใช้แฟ้มที่ตั้งที่ บัพเฉพาะที่ และบัพระยะไกล

การทดสอบก่อนการติดตั้งระบบ (หลังจากแบ่งกลุ่ม) เป็นการทดสอบระบบทีละส่วนก่อนการติดตั้งใช้งานจริง ในการกระจายแฟ้มจะเป็นการทดสอบการเข้าถึงแฟ้มระยะไกล วิธีการทดสอบจะเขียนโปรแกรมขึ้นมาชุดหนึ่ง ให้มีการเรียกใช้แฟ้มพร้อมกันหลายๆ แฟ้ม แล้วให้อ่านระเบียบจากแฟ้มเหล่านั้น

IDENTIFICATION DIVISION.

Program-id. TEST_RESPONS.

environment division.

configuration section.

source-computer MICROVAXII.

object-computer MICROVAXII.

input-output section.

file-control.

select customer-file assign to CUSTOMER

organization is indexed

access mode is dynamic

record key is CUS-ACCOUNT

alternate record key is CUS-NAME-KEY with duplicates

alternate record key is CUS-MOBILE

alternate record key is CUS-MASTER-ACCT with duplicates

file status is F-STATUS.

data division.

file section.

copy "dua0:[cat.lib]customer_rec.cpy".

working-storage section.

01 F-STATUS pic xx.

01 RESULT pic s9(9) comp.

01 START-ACCT pic 9(7).

procedure division.

declaratives.

ERROR-HANDLE section.

use after standard error procedure on input

end declaratives.

SOURCE-PROGRAM section.

BEGIN.


```

display "This Program is test response time: "
1  open input CUSTOMER-FILE allowing all
display "ENTER CUSTOMER ACCOUNT NUMER: " with no advancing
2  accept START-ACCT conversion protected size 7
move START-ACCT to CUS-ACCOUNT
3  call "lib$init_timer" giving RESULT
if RESULT is failure call "lib$stop" using by value RESULT
display "usage value before initialization:".
call "lib$show_timer" giving RESULT
if RESULT is failure call "lib$stop" using by value RESULT
4  read CUSTOMER-FILE key is CUS-ACCOUNT
display "usage value after initialization:".
5  call "lib$show_timer" giving RESULT
if RESULT is failure call "lib$stop" using by value RESULT
close CUSTOMER-FILE
stop run.

```

รูปที่ 4.18 แสดงตัวอย่างโปรแกรม ที่เรียกใช้โปรแกรมเฉพาะงานจากคลัง เพื่อทดสอบเวลาตอบสนอง

คำอธิบายการทำงานของโปรแกรมทดสอบ

โปรแกรมทดสอบที่พัฒนาขึ้นเพื่อใช้ทดสอบเวลาตอบสนองไปในการอ่านแฟ้มหนึ่งระเบียนจะใช้เวลาทั้งหมดเท่าไร โดยมีเงื่อนไขการทดสอบมีดังนี้

1. กำหนดค่าเริ่มต้น (lib\$init_timer) หลังคำสั่งภาษาโคบอลให้เปิดแฟ้ม (คำสั่ง open)
2. กำหนดค่าเริ่มต้น (lib\$init_timer) ก่อนคำสั่งภาษาโคบอลให้เปิดแฟ้ม (คำสั่ง open) จากโปรแกรมทดสอบให้เลื่อนคำสั่ง "1" มาไว้ก่อนคำสั่ง "4" หนึ่งบรรทัด
3. ใช้คำสั่งดีซีแอลกำหนดชื่อแฟ้มเชิงตรรกะ (ASSIGN) ให้โปรแกรมเรียกใช้แฟ้ม "CUSTOMER.DAT" ที่แฟ้มนี้ตั้งอยู่ที่บัพเดียวกับโปรแกรมทดสอบดำเนินงาน (local node)

```
$ assign dua0:[cat.dat]customer.dat CUSTOMER
```

```
$ Run TEST_RESPONS
```

4. ใช้คำสั่งดีซีแอลกำหนดชื่อแฟ้มเชิงตรรกะ (ASSIGN) ให้โปรแกรมเรียกใช้แฟ้ม “CUSTOMER.DAT” ที่แฟ้มนี้ตั้งอยู่ต่างบัพกับโปรแกรมทดสอบดำเนินงาน (remote node)

```
$ SHOW NETWORK
```

```
VAX/VMS network for local node 9.1 CATRAD on DD-MMM-YYYY hh:mm:ss.00
```

```
$ assign CATBUS::dua0:[cat.dat]customer.dat CUSTOMER
```

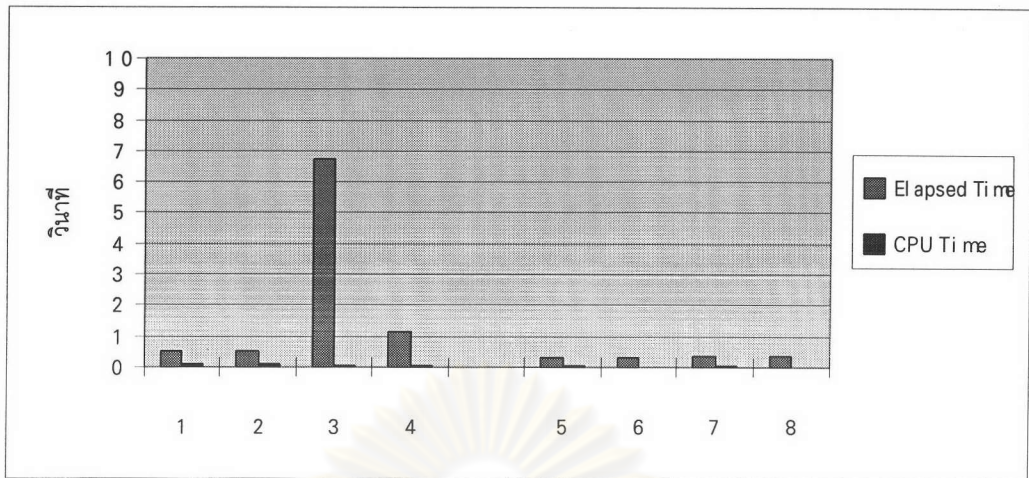
```
$ run TEST_RESPONS
```

การทำงานของโปรแกรม

1. เริ่มต้นโปรแกรมทดสอบจะเปิดแฟ้มข้อมูลชื่อ “CUSTOMER”
2. รับค่าคีย์ของระเบียบที่ต้องการอ่าน
3. เริ่มนับเวลา
4. อ่านระเบียบจากแฟ้มตามค่าคีย์ที่สั่งไว้
5. แสดงเวลาที่โปรแกรมใช้ในการอ่านระเบียบแบบดัชนี หนึ่งระเบียบ

เริ่มนับเวลาก่อนคำสั่ง OPEN แฟ้ม บัพที่ตั้งแฟ้ม		ELAPSED (วินาที)	CPU (วินาที)	BUFIO (ครั้ง)	DIRIO (ครั้ง)	FAULTS (ครั้ง)
บัพเฉพาะที่	1. ดำเนินงานครั้งที่ 1	0.54	0.11	5	10	8
	2. ดำเนินงานครั้งที่ 2	0.51	0.12	5	10	8
บัพระยะไกล	3. ดำเนินงานครั้งที่ 1	6.74	0.06	13	0	14
	4. ดำเนินงานครั้งที่ 2	1.16	0.04	13	0	24
เริ่มนับเวลาหลังคำสั่ง OPEN แฟ้ม						
บัพเฉพาะที่	5. ดำเนินงานครั้งที่ 1	0.33	0.05	3	6	13
	6. ดำเนินงานครั้งที่ 2	0.32	0.02	3	6	5
บัพระยะไกล	7. ดำเนินงานครั้งที่ 1	0.35	0.05	5	0	13
	8. ดำเนินงานครั้งที่ 2	0.36	0.02	5	0	5

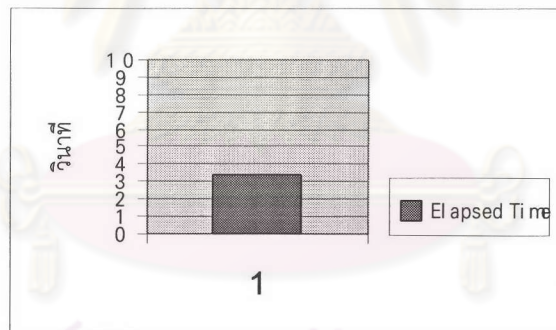
ตารางที่ 4.8 แสดงการเปรียบเทียบเวลาตอบสนองของการอ่านระเบียบแฟ้มดัชนี ระหว่างบัพเฉพาะที่กับบัพระยะไกล



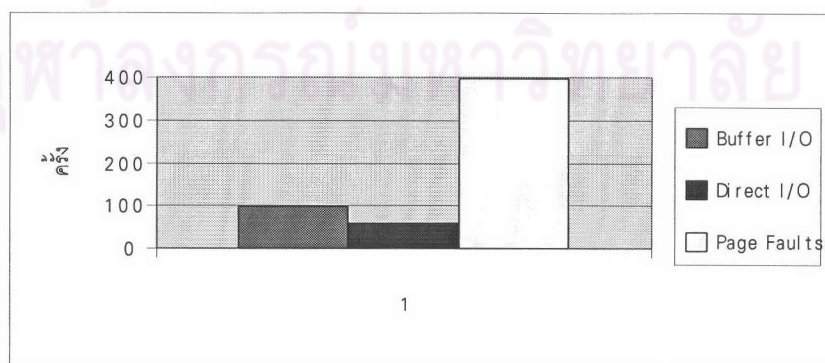
รูปที่ 4.19 แสดงเวลาตอบสนองของการเข้าถึงแฟ้มเฉพาะที่ และระยะไกลตามตารางที่ 4.8

บัพที่ติดตั้งแฟ้มที่ DECnet ให้บริการการเข้าถึงแฟ้มระยะไกล	Elapsed (วินาที)	BUFIO (ครั้ง)	DIRIO (ครั้ง)	FAULTS (ครั้ง)
บัพระยะไกล	3.39	96	60	397

ตารางที่ 4.9 แสดงการใช้ทรัพยากรระบบของ เดคเน็ต ที่ให้บริการการเข้าถึงแฟ้มระยะไกล



รูปที่ 4.20 แสดงเวลาที่เดคเน็ตเข้าใช้ ซีพียูเพื่อบริการการเข้าถึงแฟ้มระยะไกล



รูปที่ 4.21 แสดงจำนวนครั้งที่เดคเน็ตเข้าใช้ อุปกรณ์รับเข้า-ส่งออก เพื่อบริการการเข้าถึงแฟ้มระยะไกล

จากการดำเนินงานโปรแกรมทดสอบ แล้วบันทึกค่าที่โปรแกรมเฉพาะงานจากคลัง ได้แสดงค่าที่โปรแกรมทดสอบ ใช้เวลา และใช้ทรัพยากรระบบ ลงในตารางที่ 4.8 เมื่อพิจารณาการดำเนินงานลำดับที่ 3 จะเห็นว่าใช้เวลา 6.74 วินาที เป็นการใช้เวลาในการอ่านระเบียบมากที่สุด เนื่องจากเป็นการจับเวลาตั้งแต่เปิดแฟ้มเฉพาะที่เป็นแฟ้มระยะไกลจนถึงการอ่านระเบียบแบบ ดัชนีได้เรียบร้อย แต่เมื่อพิจารณาการดำเนินการลำดับที่ 4 เป็นการดำเนินงานด้วยเงื่อนไขการ ทดสอบเดียวกับลำดับที่ 3 ซึ่งเป็นการดำเนินการถัดมาในเวลาไล่เลี่ยกัน แต่ใช้เวลาเพียง 1.16 วินาทีซึ่งน้อยกว่าถึง 5.58 วินาที

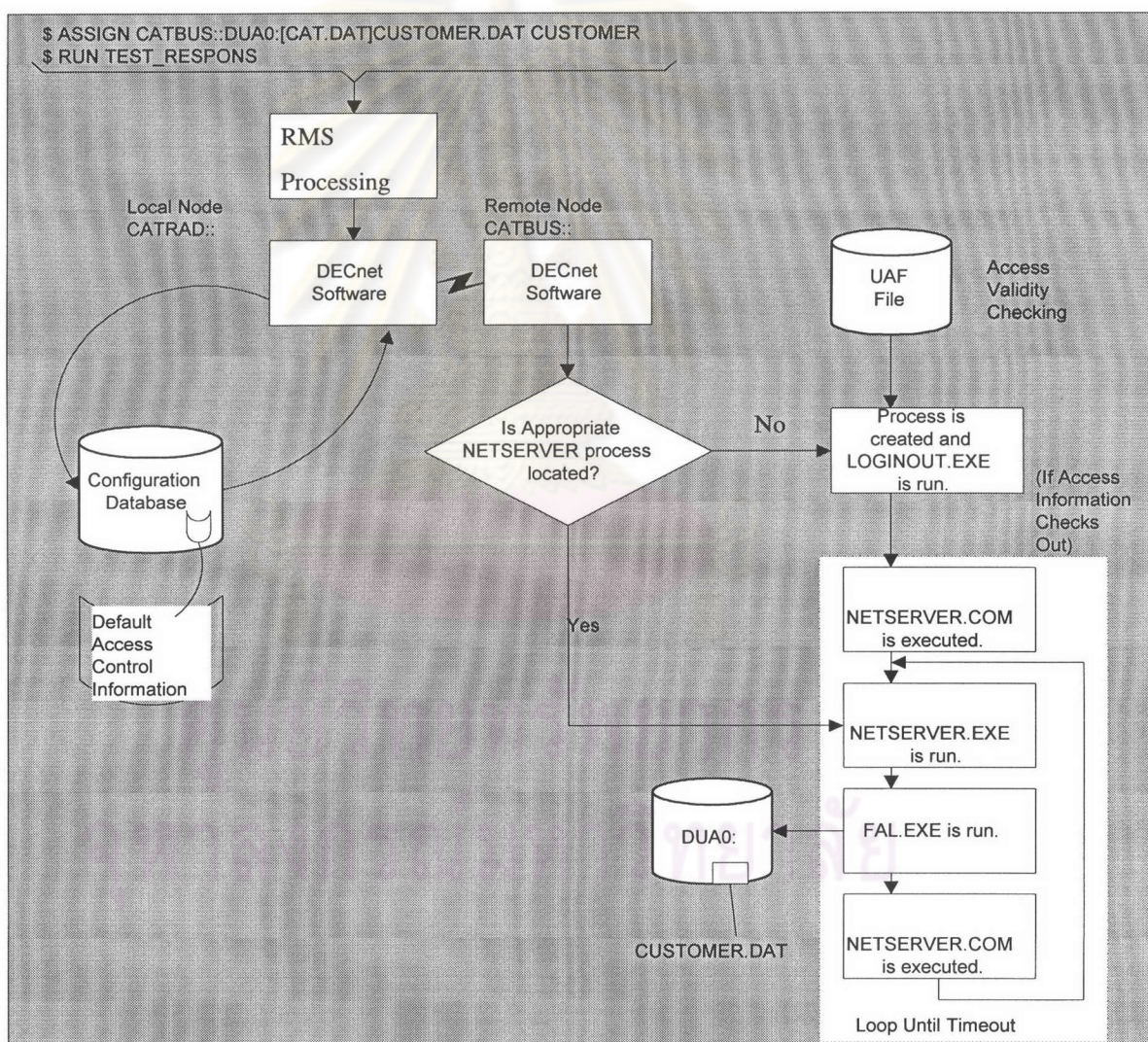
พิจารณาการทดสอบการดำเนินงานโปรแกรมทดสอบในเงื่อนไขลำดับที่ 5 ถึง 8 เป็นการจับเวลาหลังจากการเปิดแฟ้มแล้วจึงเริ่มจับเวลา จนถึงอ่านระเบียบได้เรียบร้อยจะเห็นว่าใช้ เวลาใกล้เคียงกันมาก จนเกือบไม่ต่างกัน

จะเห็นได้ว่าเวลาที่ใช้ในการเปิดแฟ้มระยะไกลครั้งแรก จะมากที่สุด เมื่อได้พิจารณา ลำดับการทำงานของการทำงานเข้าถึงแฟ้มระยะไกล ตามรูป 4.22 จะเห็นว่ามีลำดับการทำงานหลายขั้น ตอนในการดำเนินงานโปรแกรมทดสอบเวลาตอบสนองการทำงานเข้าถึงระเบียบ “TEST_RESPONS.EXE” ในเงื่อนไขการทดสอบลำดับที่ 3, 4, 7 และ 8 เป็นการเข้าถึงแฟ้มระยะไกลจะมีลำดับการดำเนินการดังนี้

1. เมื่อดำเนินงานโปรแกรมทดสอบถึงคำสั่งเปิดแฟ้ม การบริการจัดการแฟ้มจะเข้าไปดูตารางการใช้ชื่อแฟ้มเชิงตรรกะพบว่าต้องเข้าถึงแฟ้มที่บัพระยะไกลชื่อบัพ “CATBUS” ระบบ การจัดการระเบียบอาเอมเอส (RMS) จะขอบริการจากเดคเน็ต
2. เดคเน็ตจะทำการเชื่อมโยงโดยสร้างการเชื่อมโยงทางตรรกะ (logical link) ขึ้นมา หนึ่งวงจร (circuit)
3. เดคเน็ตจะทำการตรวจสอบว่าผู้ใช้ที่ใช้วงจรมีกระบวนการ “NETSERVER” หรือยังถ้ายังไม่มีจะต้องผ่านโปรแกรมลงบันทึกเข้าออก (LOGINOUT.EXE) ก่อนเพื่อตรวจสอบสิทธิ การเข้าใช้ระบบก่อน
4. เมื่อผ่านการตรวจสอบจะดำเนินงานสร้างกระบวนการ โดยเข้าไปใช้แฟ้มชุด คำสั่งชื่อ “NETSERVER.COM” สร้างกระบวนการขึ้นมาเพื่อมาบริการการเชื่อมโยงวงจรมี
5. แฟ้มชุดคำสั่ง “NETSERVER.COM” จะดำเนินการโปรแกรม “NETSERVER.EXE” เพื่อมาบริการการเชื่อมโยงวงจรมี
6. เมื่อบัพเฉพาะที่ร้องขอการเข้าถึงแฟ้ม กระบวนการ “NETSERVER.EXE” จะ ดำเนินการโปรแกรม FAL.EXE (File Access Listener) ทำหน้าที่การเข้าถึงแฟ้ม

7. เมื่อการเข้าถึงแฟ้มไม่ว่าจะเป็นการเปิดแฟ้ม อ่านระเบียบ เขียนระเบียบ ปรับปรุงระเบียบ หรือปิดแฟ้ม สำหรับแฟ้มทุกชนิดทุกรูปแบบ โปรแกรมเอฟเอแอล (FAL.EXE) จะทำงานจนกว่าจะยกเลิกการเชื่อมต่อ

8. ถึงแม้ว่าการเชื่อมต่อจะยกเลิกแล้วก็ตามแต่กระบวนการ "NETSERVER.COM" จะยังคงดำเนินงานอยู่จะกว่าจะหมดเวลา (time out) ประโยชน์ก็คือถ้ามีการเชื่อมต่อจริงจากบัพระยะไกลเข้ามาอีกกระบวนการนี้สามารถเข้าบริการได้ทันทีไม่ต้องรอสร้างกระบวนการใหม่ขึ้นมารองรับ



รูปที่ 4.22 กระบวนการเข้าถึงแฟ้มระยะไกลของเดคเน็ต

จะเห็นได้ว่าการเข้าถึงแฟ้มระยะไกลเริ่มแรกจะเกิดกระบวนการ 8 ขั้นตอนทำให้ใช้เวลาในการดำเนินการมากตามการทดสอบลำดับที่ 3 และการใช้ทรัพยากรที่บัพระยะไกลแสดงไว้ตามตารางที่ 4.8

ส่วนการทดสอบลำดับที่ 4 ใช้เวลาในการดำเนินการน้อยกว่าเนื่องจากลดกระบวนการได้หลายขั้นตอนดังนี้

1. เมื่อดำเนินงานโปรแกรมทดสอบถึงคำสั่งเปิดแฟ้ม การบริการจัดการแฟ้มจะเข้าไปดูตารางการใช้ชื่อแฟ้มเชิงตรรกะพบว่าต้องเข้าถึงแฟ้มที่บัพระยะไกลชื่อบัพ "CATBUS" ระบบการจัดการระเบียบอาเอมเอสจะขอบริการจากเดคเนต
2. เดคเนตจะทำการเชื่อมโยงโดยสร้างการเชื่อมโยงทางตรรกะ (logical link) ขึ้นมาหนึ่งวงจร (circuit)
3. เดคเนตจะทำการตรวจสอบว่าผู้ใช้ที่ใช้วงจรนี้มีกระบวนการ "NETSERVER" หรือยัง แต่เนื่องจากกระบวนการ "NETSERVER.EXE" ของผู้ใช้คนเดิมยังคงดำเนินงานอยู่จึงไม่ต้องตรวจสอบสิทธิการเข้าใช้อีก
4. เมื่อบัพเฉพาะที่ร้องขอการเข้าถึงแฟ้ม กระบวนการ "NETSERVER.EXE" จะดำเนินการโปรแกรมเอฟเอแอล (FAL.EXE หรือ File Access Listener) ทำหน้าที่การเข้าถึงแฟ้ม

การทดสอบลำดับที่ 4 จึงมีขั้นตอนดำเนินงานเพียง 4 ขั้นตอน ส่วนการทดสอบลำดับที่ 7 และ 8 ใช้เวลาในการดำเนินการเท่ากับการเข้าถึงแฟ้มที่บัพเฉพาะที่ เนื่องจากมีขั้นตอนการบริการการเข้าถึงระเบียบจากกระบวนการของโปรแกรม เอฟเอแอล (FAL.EXE) เพียงขั้นตอนเดียวแล้วส่งระเบียบกลับมาให้กระบวนการ "TEST_RESPONS" ที่บัพเฉพาะที่ผ่านวงจรเชื่อมโยงทางตรรกะของเดคเนตที่เชื่อมต่อไว้อยู่แล้วเท่านั้นที่ การดำเนินงานที่เข้าถึงแฟ้มที่บัพเฉพาะที่ไม่มี

4.5.2 การทดสอบการรองรับภาระงาน และเวลาตอบสนอง

การทดสอบโดยจำลองภาระงานให้กับบัพจะต้องเขียนโปรแกรมให้มีลักษณะใกล้เคียงกับการใช้งานจริงตัวอย่างของโปรแกรมจะคล้ายกับโปรแกรมที่ผ่านมา ต่างกันตรงมีการเรียกใช้แฟ้มหลายแฟ้มตามกลุ่มระบบงานดังรูปที่ 4.13 โดยการเขียนโปรแกรมจะอ่านระเบียบจากแฟ้มหลัก แล้วใช้เขตข้อมูลที่เป็นคีย์ของอีกแฟ้มหนึ่ง (foreign key) นำไปใช้อ่านระเบียบต่อๆกันไปจนครบแฟ้มในกลุ่ม แล้วเริ่มระเบียบใหม่ของแฟ้มหลักทั้งหมดจะดำเนินงานเป็นลูป เพื่อจำลองว่าเป็นภาระงาน (work load) ที่เข้าไปดำเนินงานในระบบ

เมื่อได้จำลองภาระงานของแต่ละระบบ จะมีโปรแกรมอีกชุดหนึ่งเรียกใช้เพิ่มเป็นกลุ่มตามระบบงาน แต่เป็นการดำเนินงานแบบโต้ตอบใช้สำหรับทดสอบเวลาตอบสนองแบบเชิงโต้ตอบโดยในโปรแกรมจะมีรันไทม์ไลบรารีรูทีนของอาเอ็มเอส (RMS) ชื่อ “LIB\$INIT_TIMER” และ “LIB\$SHOW_TIMER” สำหรับแสดงเวลาที่ใช้ไปทั้งหมดในการเข้าถึงระเบียบ ในหนึ่งระเบียบ การทดสอบกำหนดเงื่อนไขดังนี้

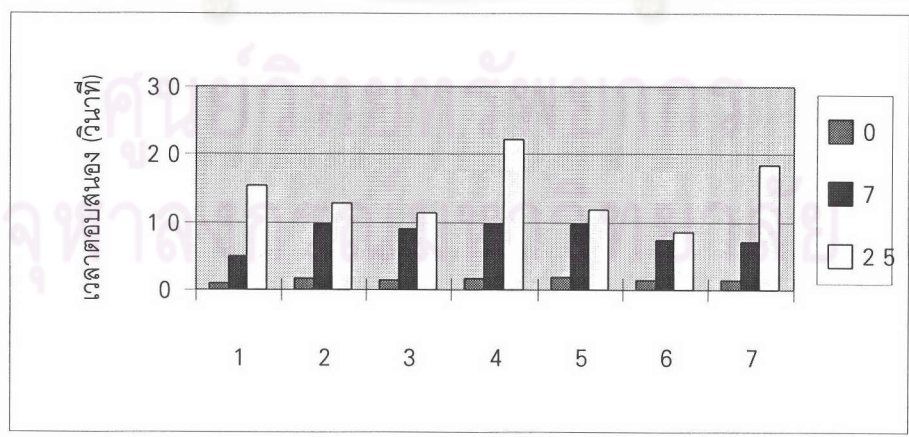
1. ทดสอบเวลาตอบสนองในการดำเนินการในระบบแบบรูปศูนย์ (Centralized System) รูปแบบเดิม

1.1 ดำเนินงานโปรแกรมแบบโต้ตอบที่ละชุด ของกลุ่มงาน เพื่อทดสอบเวลาตอบสนองของกลุ่มงานในภาวะที่ไม่มีภาระงาน แล้วบันทึกเวลาที่ใช้

1.2 ดำเนินงานโปรแกรมจำลองภาระงานในระบบ 7 และ 25 กระบวนการตามลำดับ จากนั้นดำเนินงานโปรแกรมแบบโต้ตอบที่ละชุดของกลุ่มงานจำนวน 5 ครั้ง เพื่อทดสอบเวลาตอบสนองของกลุ่มงาน แล้วคำนวณเวลาเฉลี่ยของทั้ง 5 ครั้ง แล้วบันทึกผลตาราง

เวลาตอบสนองการทดสอบโปรแกรม ที่เป็นแบบแฟ้มรวมศูนย์ (ELAPSED) (วินาที)							
จำนวนภาระงาน (work Load)	ระบบทะเบียนวิทยุ	ระบบเปิดบริการ	ระบบเปลี่ยนแปลงการใช้	ระบบคิดค่าใช้จ่าย	ระบบจัดเตรียม,พิมพ์ใบแจ้งหนี้	ระบบรับชำระค่าใช้บริการ	ระบบบัญชีส่วนกลาง, ภูมิภาค
0	0.9	1.77	1.32	1.7	1.92	1.44	1.37
7	4.97	9.78	8.86	9.58	9.77	7.36	7.19
25	15.32	12.7	11.33	22.13	11.77	8.62	18.42

ตารางที่ 4.10 แสดงเวลาตอบสนองที่ได้จากการทดสอบโปรแกรมก่อนการกระจายแฟ้ม



รูปที่ 4.23 แสดงเวลาตอบสนองในการเข้าถึงแฟ้มเฉพาะที่ของกลุ่มงานในขณะที่มีการะงานเพิ่มขึ้น

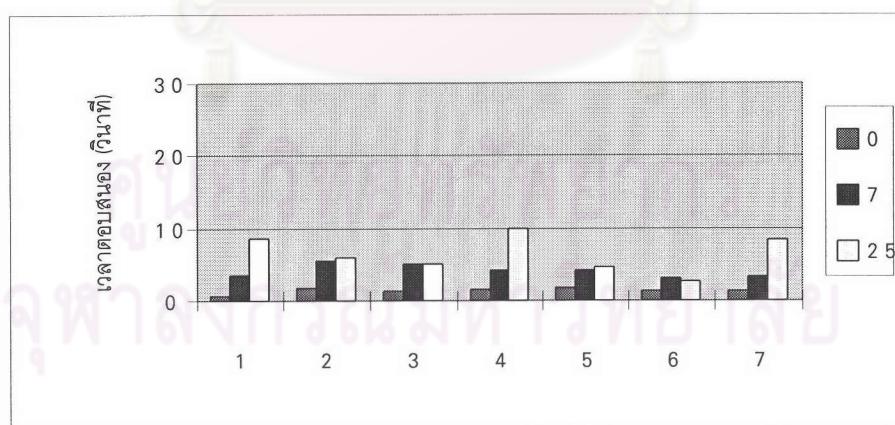
2. ทดสอบเวลาตอบสนอง ในการดำเนินการระบบแบบ กระจายเพิ่ม (Distributed File) ตามที่ได้ออกแบบไว้

2.1 ดำเนินงานโปรแกรมแบบโต้ตอบที่ละชุด ของกลุ่มงานที่บันทึกที่กำหนดไว้ ตามกลุ่มงานที่ได้ออกแบบ เพื่อทดสอบเวลาตอบสนองของกลุ่มงานในสถานะที่ไม่มีภาระงาน

2.2 ดำเนินงานโปรแกรมจำลองภาระงานในระบบ 7 และ 25 กระบวนการ ตามลำดับ จากนั้นดำเนินการโปรแกรมแบบโต้ตอบที่ละชุดของกลุ่มงานจำนวน 5 ครั้ง เพื่อทดสอบเวลาตอบสนองของกลุ่มงาน แล้วคำนวณเวลาเฉลี่ยของทั้ง 5 ครั้ง แล้วบันทึกผลตาราง

เวลาตอบสนองการทดสอบโปรแกรมหลังออกแบบการกระจายเพิ่ม (ELAPSED) (วินาที)							
จำนวนภาระงาน (Load)	ระบบ ทะเบียน วิทยุ	ระบบเปิด บริการ	ระบบ เปลี่ยนแปลงการใช้	ระบบคิดค่า ใช้บริการ	ระบบจัด เตรียม, พิมพ์ใบแจ้งหนี้	ระบบรับ ชำระค่าใช้จ่าย บริการ	ระบบบัญชี ส่วนกลาง, ภูมิภาค
0	0.71	1.71	1.27	1.53	1.73	1.34	1.28
7	3.51	5.55	5.04	4.30	4.22	3.01	3.22
25	8.70	5.94	5.09	9.99	4.53	2.64	8.44

ตารางที่ 4.11 แสดงเวลาตอบสนองที่ได้จากการทดสอบโปรแกรม หลังจากการกระจายเพิ่มแล้ว



รูปที่ 4.24 แสดงเวลาตอบสนองในการเข้าถึงแฟ้มระยะไกลในขณะมีภาระงานเพิ่มขึ้น

เมื่อได้พิจารณาเวลาตอบสนองหลังจากได้กระจายเพิ่มแล้ว จะเห็นได้ว่าขณะที่ไม่มีภาระงานเวลาตอบสนองของกระบวนการแบบโต้ตอบของระบบแบบรวมศูนย์จะเร็วกว่า แต่เมื่อมีภาระงานมากขึ้น กระบวนการที่ดำเนินการแบบกระจายเพิ่ม เวลาตอบสนองจะเร็วกว่า

การทดสอบเปรียบเทียบก่อน และหลังจากการใช้งานจริง

การทดสอบขณะใช้งานจริงไม่สามารถทดสอบแบบการใช้การจำลองภาระงาน เนื่องจากเวลาตอบสนองไม่สามารถทำได้ เพราะการใช้รันไทม์ไลบรารีที่รันจะต้องแก้ไขโปรแกรม และกระทบการปฏิบัติงานของผู้ใช้ และภาระงานของระบบงานจริงไม่คงที่ไม่สามารถให้ทดสอบจริงได้ เนื่องจากค่าที่วัดได้ไม่แน่นอนไม่สามารถใช้อ้างอิงได้

การเก็บข้อมูลของระบบ

วิธีการทดสอบวิเคราะห์สมรรถนะของการดำเนินงานโปรแกรมประยุกต์บนระบบทำได้ โดยใช้โปรแกรมอรรถประโยชน์ที่มีมาพร้อมระบบปฏิบัติการทำการเก็บรวบรวมข้อมูล ซึ่งมีวิธีการเก็บ 2 แบบคือ

1. เก็บในเชิงปริมาณ ใช้โปรแกรมอรรถประโยชน์ เอกเค้าตัง (ACCOUNTING)
2. เก็บในเชิงเทียบกับเวลาใช้โปรแกรมอรรถประโยชน์ มอนิเตอร์ (MONITOR)

เงื่อนไขในการจัดเก็บข้อมูล

กำหนดเวลาที่จัดเก็บใช้เงื่อนไขของการทำงานที่ใกล้เคียงกันมากที่สุด ในที่นี้กำหนดสัปดาห์ที่ 2 ของเดือนโดยเริ่มเก็บข้อมูลวันอาทิตย์ที่ 7 มกราคม 2539 เวลา 00:00 นาฬิกา ไปสิ้นสุดวันที่ 13 มกราคม 2539 เวลา 23:59 นาฬิกา ส่วนข้อมูลระบบหลังการติดตั้งใช้งานระบบที่ออกแบบใหม่แล้วกำหนดวันที่ 10 มีนาคม 2539 เวลา 00:00 นาฬิกา ไปสิ้นสุดวันที่ 16 มีนาคม 2539 เวลา 23:59 นาฬิกา

วิธีการเก็บข้อมูลใช้เอกเค้าตัง (ACCOUNTING)

\$ SET ACCOUNT/ENBLE

:

\$ ACCOUNTING -

/REPORT=(PROCESSOR,BUFFERED_IO,DIRECT_IO,EXECUTION,FAULTS, PAGE_READS) -

/SINCE=DD-MMM-YYYY:hh:mm:ss/BEFORE=DD-MMM-YYYY:hh:mm:ss -

/ACCOUNT=(ชื่อบัญชีที่กำหนดไว้ตอนเปิดบัญชีให้ผู้ใช้เข้าใช้ระบบ) -

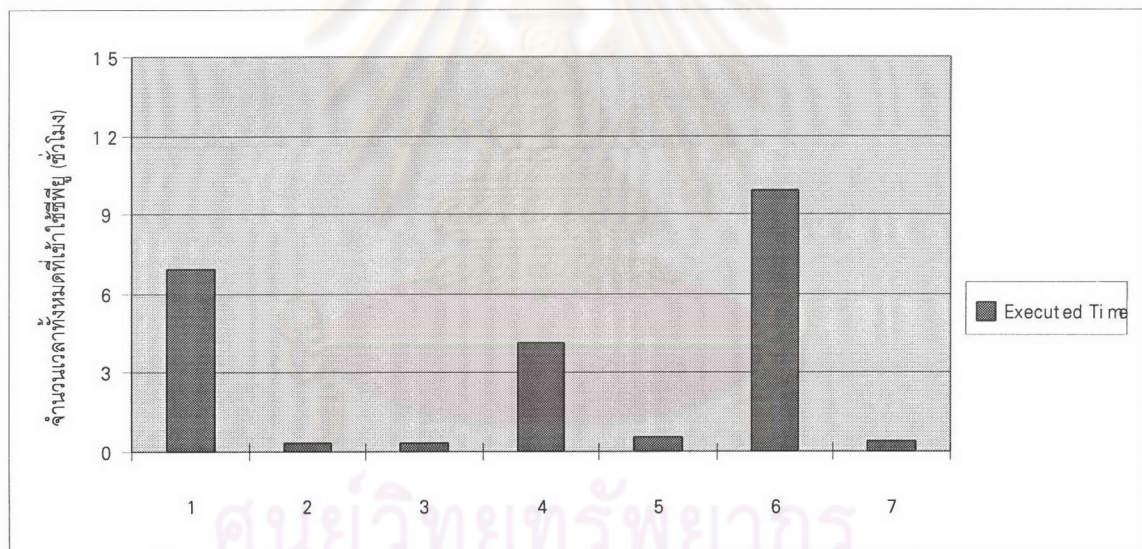
/OUTPUT=(ชื่อแฟ้มที่จะบันทึกไว้สำหรับใช้พิมพ์) -

/SUMMARY=(ACCOUNT, NODE) (รายงานสรุปยอดรวมตามกลุ่มที่กำหนด)

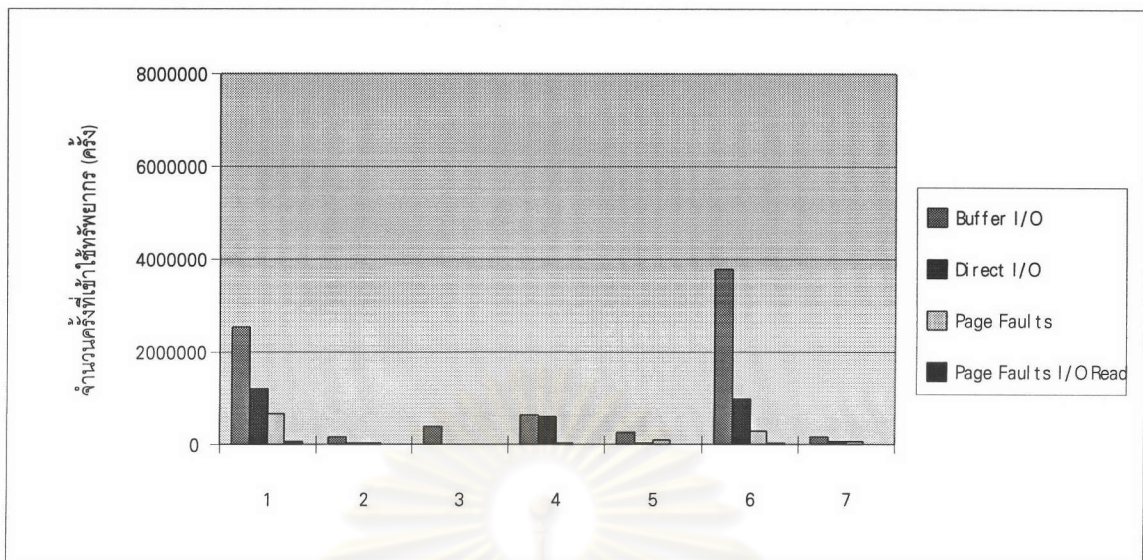
เพิ่มรายงานเข้าพุทจะมีสารสนเทศ แสดงปริมาณการใช้ทรัพยากรระบบ สรุปยอดรวมแบ่งตามกลุ่มงานเพื่อให้สามารถวิเคราะห์ปริมาณงานที่แต่ละกลุ่มดำเนินงาน

		ปริมาณการใช้ทรัพยากรก่อนการกระจายเพิ่ม			สัปดาห์ที่ 2 ของเดือน	
จากวันที่ 7 มกราคม 2539 เวลา 00:00 นาฬิกา		การเข้าใช้ระบบของกลุ่มงาน แยกตามกลุ่มงาน			ถึงวันที่ 13 มกราคม 2539 เวลา 23:59 นาฬิกา	
กลุ่มงาน	เวลาที่ใช้ซีพียู	บัพเฟอไอโอ	ไดเร็คไอโอ	จำนวนครั้งที่ สั่ง ต้า เนิน งาน	การบกพร่องหน้า	การบกพร่องหน้าที่ต้อง อ่านจากจานแม่เหล็ก
1. เปิดบริการ	0 06:55:51.11	2540176	1194186	6677	659201	73945
2. ทะเบียน	0 00:20:43.25	147357	44821	497	37731	5194
3. ตรวจสอบ	0 00:18.30.55	373007	2321	93	6757	891
4. คิดค่าใช้	0 04:08:00.88	650125	591355	387	44669	3433
5. โบบ้างหนี้	0 00:32:51.60	267322	28636	942	103242	7555
6. รับชำระ	0 09:55:47.38	3766388	976504	4201	283335	40837
7. บัญชี	0 00:23:26.63	156177	48808	1060	69836	10307

ตารางที่ 4.12 แสดงปริมาณการใช้ทรัพยากรระบบที่ได้จากการเก็บข้อมูลก่อนการกระจายเพิ่ม



รูปที่ 4.25 แสดงปริมาณการเข้าใช้ซีพียูในหนึ่งสัปดาห์ ก่อนการกระจายเพิ่ม

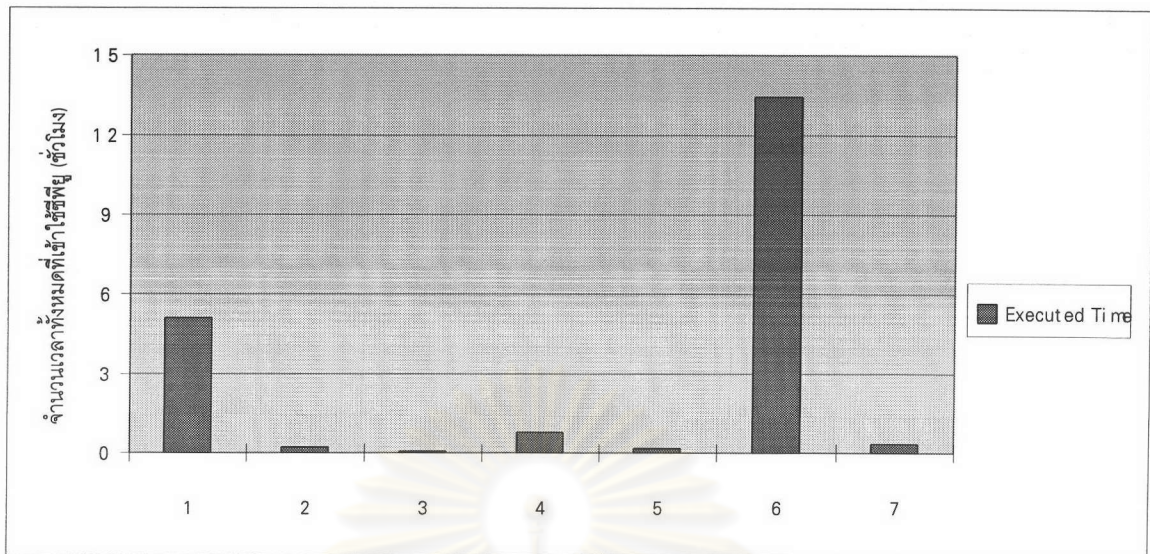


รูปที่ 4.26 แสดงปริมาณการใช้ทรัพยากรระบบก่อนการกระจายเพิ่มตามตารางที่ 4.12

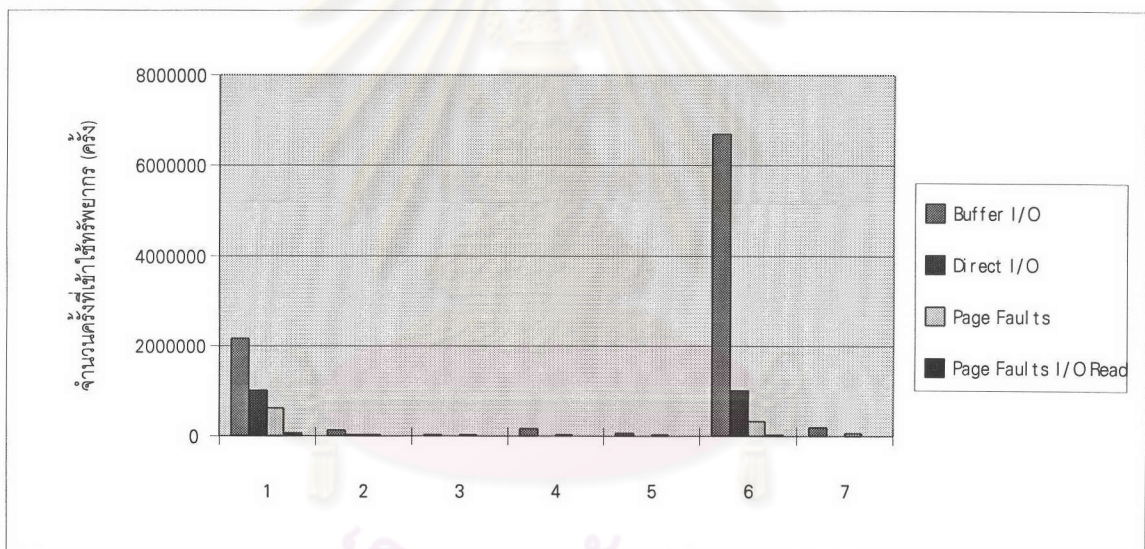
		การกระจายเพิ่ม			สัปดาห์ที่ 2 ของเดือน	
จากวันที่ 10 มีนาคม 2539 เวลา 00:00 นาฬิกา		การเข้าใช้ระบบของกลุ่มงาน แยกตามกลุ่มงาน			ถึงวันที่ 16 มีนาคม 2539 เวลา 23:59 นาฬิกา	
กลุ่มงาน	เวลาที่ใช้ซีพียู	บัฟเฟอร์ไอโอ	ไดเรคไอโอ	จำนวนครั้งที่ดำเนินงาน	การบกพร่องหน้า	การบกพร่องหน้าที่ต้องอ่านจากจานแม่เหล็ก
บัพ CATBUS						
เปิดบริการ	0 05:05:07.02	2141319	1025228	6113	614098	68083
ทะเบียน	0 00:12:51.00	115206	28001	411	31262	4440
บัพ CATRAD						
ตรวจสอบ	0 00:03:35.03	26279	2014	269	19172	2492
คิดค่าใช้	0 00:47:46.43	173782	4580	278	17195	2498
ใบแจ้งหนี้	0 00:08:42.09	57051	4637	386	29137	3803
บัพ CATACT						
รับชำระ	0 13:25:02.56	6684460	1026259	4383	335348	44108
บัญชี	0 00:21:24.85	208019	16055	855	67325	8798

ตารางที่ 4.13 แสดงปริมาณการใช้ทรัพยากรระบบ ที่ได้จากการเก็บข้อมูล

หลังจากการกระจายเพิ่มแล้ว



รูปที่ 4.27 แสดงการใช้ซีพียูแยกตามกลุ่มงานหลังการกระจายเพิ่ม ตามตารางที่ 4.13



รูปที่ 4.28 แสดงปริมาณการใช้ทรัพยากรระบบหลังการกระจายเพิ่มตามตารางที่ 4.13

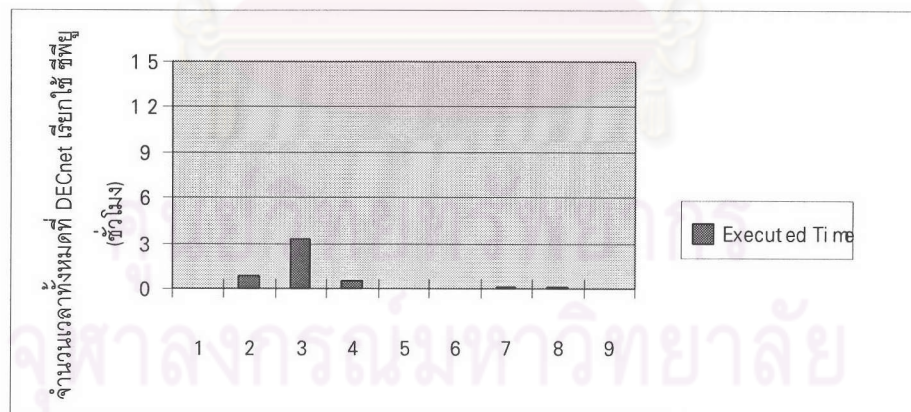
จากข้อมูลปริมาณการเข้าใช้ทรัพยากรของระบบ แสดงให้เห็นว่าปริมาณการใช้ทรัพยากรระบบ ทั้งซีพียู ไอโอ และหน่วยความจำ ของแต่ละกลุ่มงาน ก่อนการกระจายเพิ่ม และหลังจากติดตั้งระบบการกระจายเพิ่มแล้ว ปรากฏว่าปริมาณการใช้งานใกล้เคียงกัน แต่จะมีบางกลุ่มงานมีปริมาณงานเพิ่มขึ้นเช่น ระบบการรับชำระ ปริมาณการใช้ซีพียู และไอโอเพิ่มขึ้นเป็นสองเท่า เนื่องจากมีปริมาณรายการเปลี่ยนแปลงเพิ่มขึ้น

ส่วนของปริมาณของระบบที่เพิ่มเข้ามาคือ กระบวนการของเดคเนตที่ใช้ในการส่งรับรายการข้ามบัพ ที่เกิดขึ้นหลังจากมีการกระจายข้อมูลไปทั้งสามบัพ มีข้อสังเกตคือบัพที่ชื่อ

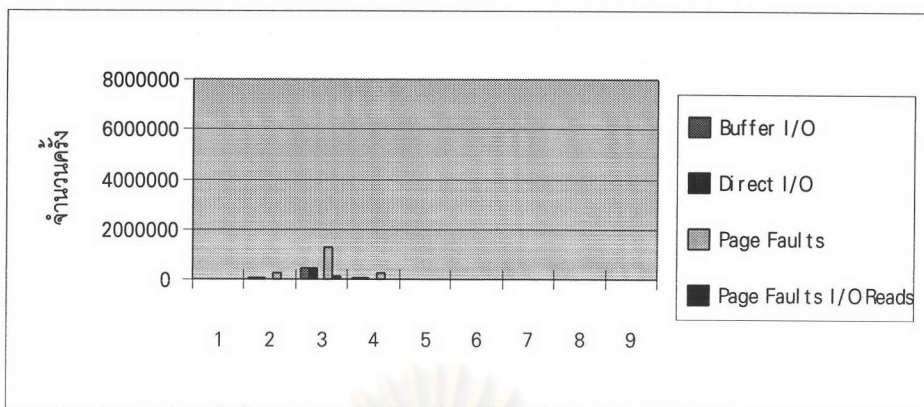
“CATBUS” จะมีปริมาณที่กระบวนการเดคเน็ตให้บริการการเข้าถึงแฟ้มระยะไกลที่ติดตั้งอยู่ที่บัพเฉพาะที่ ให้กับบัพ CATACT ที่ร้องขอเข้ามาปริมาณสูงกว่าบัพอื่น

		การดำเนินงาน และการเข้าถึงระหว่างบัพ			สัปดาห์ที่ 2 ของเดือน	
จากวันที่ 10 มีนาคม 2539 เวลา 00.00 น.		การดำเนินงานของเดคเน็ต แยกตามบัพ			ถึงวันที่ 16 มีนาคม 2539 เวลา 23.59 น.	
ชื่อบัพเฉพาะที่ /ชื่อบัพระยะไกล	เวลาที่ใช้ซีพียู (Executed Time)	บัพเฟอไอ ไอ (Buffer IO)	ไดเร็คไอ ไอ (Direct IO)	จำนวนครั้งที่ สั่งดำเนิน งาน (Image Count)	การบกพร่อง หน้า (Page Faults)	การบกพร่องหน้าที่ต้อง อ่านจากจานแม่เหล็ก (Page IO Reads)
CATBUS /CATBUS						
CATBUS /CATRAD	0 00:51:29.34	77436	47764	3789	263572	28201
CATBUS /CATACT	0 03:17.46.77	456159	472619	18573	1273771	140523
CATRAD /CATBUS	0 00:33:23.81	63112	54565	3768	258675	27204
CATRAD /CATRAD	0 00:00:18.34	448	453	36	2563	256
CATRAD /CATACT						
CATACT /CATBUS	0 00:05:22.87	23957	6348	444	31467	3184
CATACT /CATRAD	0 00:04:02.47	17594	4770	318	23040	2264
CATACT /CATACT	0 00:00:16.78	432	375	33	2199	235

ตารางที่ 4.14 แสดงปริมาณการใช้เดคเน็ตที่ได้จากการเก็บข้อมูลหลังจากการกระจายแฟ้มแล้ว



รูปที่ 4.29 แสดงจำนวนเวลาการใช้ซีพียูของเดคเน็ตทั้งหมด ที่วัดสถิติ ตามตารางที่ 4.14



รูปที่ 4.30 แสดงจำนวนครั้งที่ใช้เรียกใช้ทรัพยากรของ เดคเนต ที่วัดสถิติ ตามตารางที่ 4.14

เมื่อได้ปริมาณงานที่แต่ละกลุ่มงานเข้าใช้ทรัพยากรระบบ ก่อนและหลังการกระจาย
 เพิ่มแล้ว จะทำการตรวจวัดสถิติ ที่โปรแกรมประยุกต์เข้าใช้ทรัพยากรระบบ ในช่วงวัน และเวลา
 เดียวกัน เพื่อนำมาใช้วัดสมรรถนะของระบบก่อนและหลังการกระจายเพิ่ม

ซีพียู เปอร์เซ็นต์ของการเข้าใช้ซีพียู

การเกิดบกพร่องหน้าจำนวนครั้งต่อวินาที

อุปกรณ์รับเข้าส่งออก (I/O) จำนวนครั้งต่อวินาที

จำนวนเดทาแกรมในอีเทอเน็ตต่อวินาที

วิธีการเก็บข้อมูลใช้มอนิเตอร์ (MONITOR)

\$ MONITOR -

/RECORD=(ชื่อแฟ้มที่จะใช้เก็บข้อมูลการใช้ระบบ) -

/INTERVAL=(เวลาที่ใช้แสดงแต่ละช่วงเป็นวินาที) -

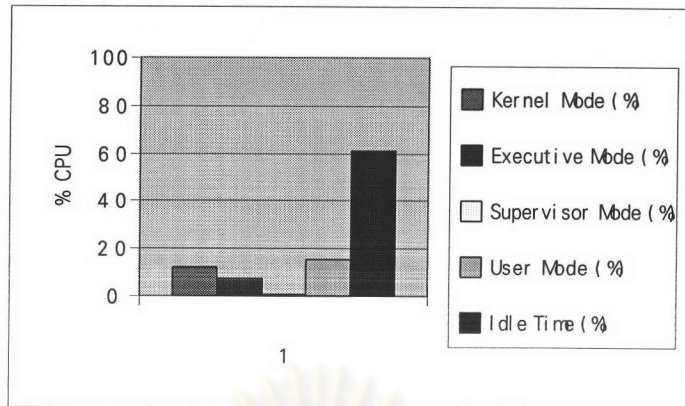
/SINCE=(วันที่ เวลาที่กำหนดในการเริ่มเฝ้าดู) -

/BEFORE=(วันที่ เวลาที่กำหนดให้สิ้นสุดการเฝ้าดู) -

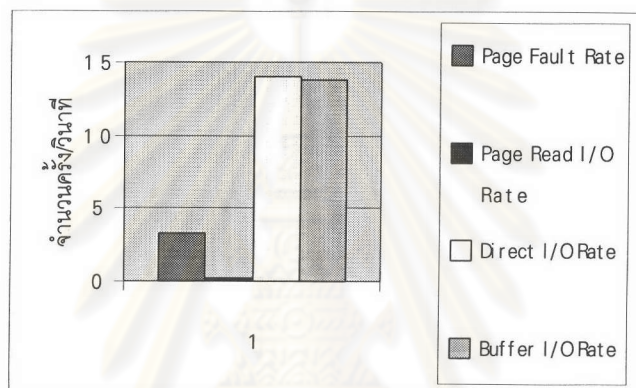
ALL_CLASSES

		สถิติระบบก่อนการกระจายเพิ่ม								
Centralize System		ก่อนการกระจายเพิ่ม						สัปดาห์ที่ 2 ของเดือน		
จากวันที่ 7 มกราคม 2539		เปอร์เซ็นต์และค่าเฉลี่ยจำนวนครั้งต่อวินาที						ถึงวันที่ 14 มกราคม 2539		
	Kernel Mode (%)	Executive Mode (%)	Supervisor Mode (%)	User Mode (%)	Idle Time (%)	Process Count (หน่วย)	Page Fault Rate	Page Read I/O Rate	Direct I/O Rate	Buffer I/O Rate
	11.8	7.5	0.8	15.6	61.3	19.0	3.3	0.2	14.0	13.8

ตารางที่ 4.15 แสดงสถิติการใช้ทรัพยากรระบบที่ได้จากการเก็บข้อมูลก่อนการกระจายเพิ่ม



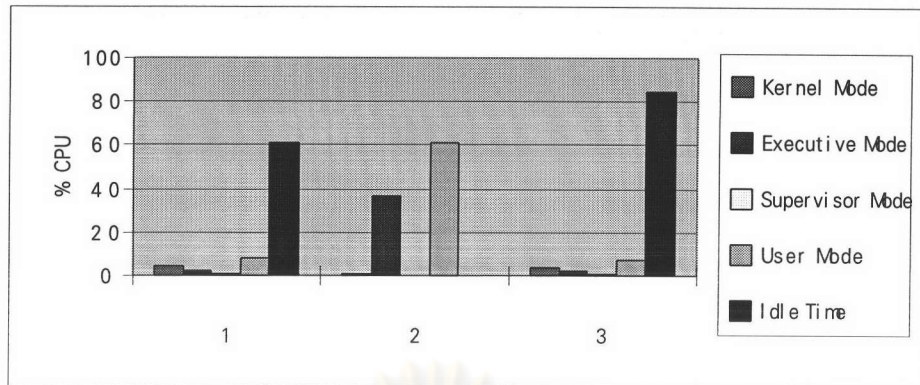
รูปที่ 4.31 แสดงเวลาการใช้ซีพียูที่เก็บข้อมูลสถิติก่อนการกระจายแฟ้ม ตามตารางที่ 4.15



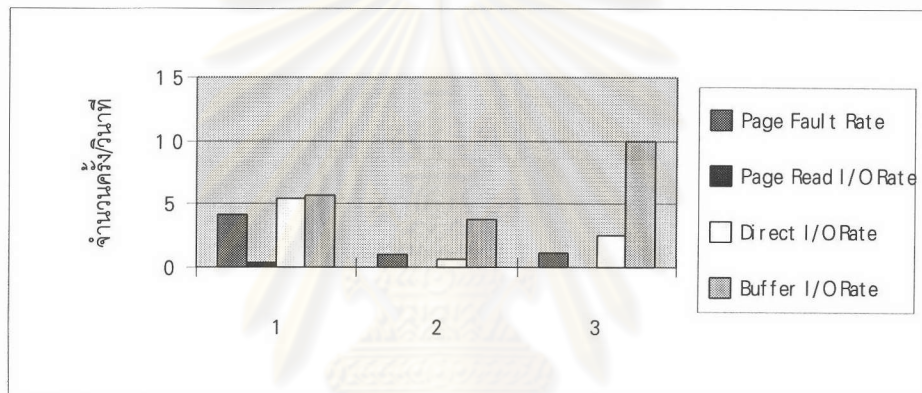
รูปที่ 4.32 แสดงสถิติการใช้ทรัพยากรระบบที่ได้จากการเก็บข้อมูลก่อนการกระจายแฟ้มตามตารางที่ 4.15

		สถิติระบบการกระจายแฟ้ม เก็บข้อมูลหลังการกระจายแฟ้ม					สัปดาห์ที่ 2 ของเดือน				
จากวันที่ 10 มีนาคม 2539 เวลา 00:00 นาฬิกา		เปอร์เซ็นต์และค่าเฉลี่ยจำนวนครั้งต่อวินาที					ถึงวันที่ 16 มีนาคม 2539 เวลา 23:59 นาฬิกา				
ชื่อแฟ้ม	Kernel Mode	Executive Mode	Supervisor Mode	User Mode	Idle Time	Process Count	Page Fault Rate	Page Read I/O Rate	Direct I/O Rate	Buffer I/O Rate	
CATBUS	4.5	2.3	0.4	8.3	61.01	18.5	4.1	0.4	5.4	5.7	
CATRAD	1.1	36.8	0.0	61.3	0.0	14.7	1.0	0.0	0.6	3.8	
CATACT	3.5	2.4	0.4	7.7	84.6	15.2	1.1	0.0	2.5	9.9	
เปอร์เซ็นต์						หน่วย	(จำนวนครั้ง/วินาที)				

ตารางที่ 4.16 แสดงสถิติการใช้ทรัพยากรระบบที่ได้จากการเก็บข้อมูลหลังกระจายแฟ้มแล้ว



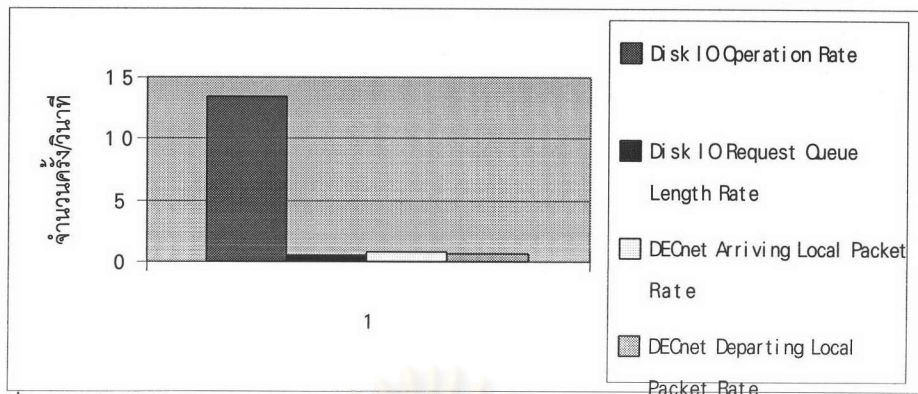
รูปที่ 4.33 แสดงจำนวนเวลาการใช้ซีพียูที่เก็บข้อมูลสถิติหลังการกระจายเพิ่ม ตามตารางที่ 4.16



รูปที่ 4.34 แสดงสถิติการใช้ทรัพยากรระบบที่ได้จากการเก็บข้อมูล หลังการกระจายเพิ่มตามตารางที่ 4.16

Centralize System		สถิติระบบก่อนการกระจายเพิ่ม			สัปดาห์ที่ 2 ของเดือน		
จากวันที่ 7 มกราคม 2539 00:00 น.		ค่าเฉลี่ยจำนวนครั้งต่อวินาที			ถึงวันที่ 13 มกราคม 2539 23:59 น.		
	Disk IO Operation Rate	Disk IO Request Queue Length Rate	DECnet Arriving Packet Rate	Local	DECnet Departing Packet Rate	Local	
	13.4	0.5		0.8		0.7	

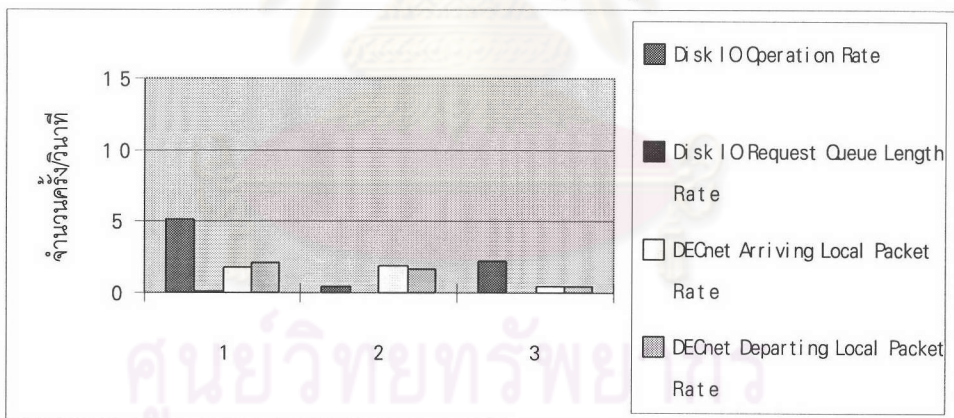
ตารางที่ 4.17 แสดงสถิติการใช้จานแม่เหล็ก และกลุ่มข้อมูลเดคเน็ต ก่อนการกระจายเพิ่ม



รูปที่ 4.35 แสดงสถิติการใช้งานแม่เหล็ก และกลุ่มข้อมูลเดคเน็ตก่อนการกระจายแฟ้ม

Distributed Files		สถิติระบบหลังการกระจายแฟ้ม		สัปดาห์ที่ 2 ของเดือน	
จากวันที่ 10 มีนาคม 2539 00:00 น.		ค่าเฉลี่ยจำนวนครั้งต่อวินาที		ถึงวันที่ 16 มีนาคม 2539 23:59 น.	
ชื่อแฟ้ม	Disk IO Operation Rate	Disk IO Request Queue Length Rate	DECnet Arriving Local Packet Rate	DECnet Departing Local Packet Rate	
CATBUS	5.2	0.1	1.8	2.1	
CATRAD	0.5	0.0	1.9	1.7	
CATACT	2.2	0.0	0.5	0.5	

ตารางที่ 4.18 แสดงสถิติการใช้งานแม่เหล็ก และกลุ่มข้อมูลเดคเน็ต หลังการกระจายแฟ้ม



รูปที่ 4.36 แสดงสถิติการใช้งานแม่เหล็ก และกลุ่มข้อมูลเดคเน็ต หลังการกระจายแฟ้ม

จากการเก็บสถิติการทำงานของระบบที่รองรับปริมาณงานจากโปรแกรมประยุกต์ต่างๆ ที่ดำเนินงานที่ใกล้เคียงกัน ทั้งก่อน และหลังการกระจายแฟ้ม ตามที่แสดงไว้ในตารางที่ 4.12 และ ตารางที่ 4.13 เป็นปริมาณงานในช่วงสัปดาห์ที่ 2 ของเดือนมีนาคมจะเพิ่มขึ้นเมื่อเทียบกับเดือน มกราคมในช่วงเวลาเดียวกันในส่วนของระบบการรับชำระ ส่วนระบบอื่น ปริมาณงานจะใกล้เคียงกันไม่ต่างกันมาก

เมื่อมาพิจารณาสมรรถนะ ที่รองรับปริมาณงานของทั้งสองช่วงแล้ว จะเห็นได้ว่า สมรรถนะหลังจากการกระจายเพิ่มข้อมูล ที่กระจายอยู่ที่สามบัพจะดีกว่า โดยดูจากแถวคอยของการเข้าใช้จานแม่เหล็ก ซึ่งส่วนใหญ่ของระบบที่ใช้ข้อมูลมากๆ จานแม่เหล็กจะเป็นสาเหตุที่สมรรถนะลดลง เนื่องจากการรอเข้าใช้ ซึ่งก่อนการกระจายเพิ่มแถวคอยจะมีปริมาณ 0.5 ครั้งต่อวินาที เมื่อเทียบกับกระจายเพิ่มแล้ว จะไม่มีแถวคอย ยกเว้นบัพที่ชื่อ CATBUS จะต้องให้บริการการเข้าถึงเพิ่มระยะไกลจากอีกสองบัพ จากที่ได้ออกแบบการกำหนดที่ตั้งเพิ่มไว้ ซึ่งก็น้อยกว่ามากมีเพียง 0.1 ครั้งต่อวินาที

หมายเหตุ ในส่วนระบบงานคิดค่าบริการ และพิมพ์ใบแจ้งหนี้มีภาระงานไม่มาก เกิดจากการนำระบบงานนี้ไปดำเนินงานที่เครื่องขนาดใหญ่เนื่องจากปริมาณรายการข้อมูลมีปริมาณมาก แต่ยังใช้บัพ CATRAD นี้เป็นส่วนของการเตรียมข้อมูล และใช้ในการตรวจสอบการใช้ เป็นงานที่ต้องใช้อุปกรณ์บันทึกเทปบ่อยครั้งซึ่งเป็นอุปกรณ์ที่ใช้เวลาของซีพียูสูงมาก ซึ่งเป็นบัญชีผู้ใช้อีกระบบหนึ่งปริมาณงานที่แสดงไว้จึงไม่มากแต่สถิติการใช้ซีพียูจะสูงดังตารางที่ 4.16

4.6 โปรแกรมอรรถประโยชน์ (Utilities Program) ที่ใช้ในการรวบรวม บันทึก และทำรายงานสถิติการใช้ทรัพยากรระบบ

4.6.1 แอคเคานท์ติ้ง “ACCOUNTING”

ใช้คำสั่งนี้เพื่อให้ระบบแสดงสารสนเทศเกี่ยวกับปริมาณการ การเข้าใช้ทรัพยากรของระบบของผู้ใช้ ซึ่งมีหลายเงื่อนไขที่จะกำหนดให้แสดง

\$ ACCOUNTING

/REPORT	รายงานที่ให้เลือกแสดง
Processor	แสดงเวลาที่เข้าใช้ ซีพียู
Buffer IO	แสดงจำนวนครั้งที่ใช้อุปกรณ์ที่ต้องใช้ผ่านบัพเฟอร์เช่นเทอร์มินัล หรือเครื่องพิมพ์
Direct IO	แสดงจำนวนครั้งที่ใช้อุปกรณ์โดยตรง เช่นจานแม่เหล็ก หรือเทปแม่เหล็ก
Image Count	จำนวนของกระบวนการที่สั่งให้ดำเนินงานโดยโปรเซสที่เข้าใช้ระบบ
Page Faults	จำนวนครั้งของการเกิดการบกพร่องหน้า ทั้งการบกพร่องที่จง

หน่วยความจำ 512 ไบท์ (Soft Page Faults) และการบกพร่องหน้าที่ ถูกสับค่า (Swap) ไปเก็บไว้ที่จานแม่เหล็กชั่วคราว เนื่องจากหน่วย ความจำไม่พอ (Hard Page Faults)

Page IO Reads	จำนวนครั้งของการอ่านหน้าที่ถูกสับค่าไปเก็บไว้ที่จานแม่เหล็ก (Hard Page Faults)
/SINCE	ตามด้วยวันที่เวลาที่ต้องการให้เริ่มรวบรวมสถิติ
/BEFORE	ตามด้วยวันที่เวลาที่ต้องการรวบรวมสถิติก่อนวันเวลาที่กำหนด
/NODE	ตามด้วยชื่อบัพที่ต้องการให้รวบรวมสถิติ
/ACCOUNT	ตามด้วยชื่อบัญชีผู้ใช้ที่ต้องการให้รวบรวมสถิติ
/SUMMARY	แสดงยอดรวมตามที่ต้องการ
ACCOUNT	แสดงยอดรวมตามชื่อบัญชีผู้ใช้ที่ระบุ
NODE	แสดงยอดรวมตามชื่อบัพที่กำหนด

4.2 มอนิเตอร์ "MONITOR"

ใช้คำสั่งนี้เพื่อให้ระบบแสดงสารสนเทศเกี่ยวกับสถิติการเข้าใช้ทรัพยากรของระบบ ทั้งหมด ซึ่งมีหลายเงื่อนไขที่จะกำหนดให้แสดง

\$ MONITOR	
/RECORD	ตามด้วยชื่อแฟ้ม เพื่อบันทึกข้อมูลสถิติเป็นแฟ้มเก็บไว้อ้างอิง
/INTERVAL	ช่วงเวลาเก็บข้อมูล (วินาที)
/INPUT	ตามด้วยชื่อแฟ้มเพื่อใช้แฟ้มข้อมูลที่บันทึกไว้มาเป็นข้อมูลเข้า ใช้ เฝ้าดูสถิติย้อนหลัง
/BEGIN	ตามด้วยวันที่เวลาที่ต้องการให้เริ่มรวบรวมสถิติ
/END	ตามด้วยวันที่เวลาที่ต้องการรวบรวมสถิติก่อนวันเวลาที่กำหนด
ALL_CLASSES	เฝ้าดูสถิติของระบบทุกชนิด
SYSTEM	เฝ้าดูสถิติการทำงานภายในของระบบปฏิบัติการ
DISK	เฝ้าดูสถิติการทำงานของจานแม่เหล็ก
PAGE	เฝ้าดูสถิติการทำงานของหน่วยความจำ
IO	เฝ้าดูสถิติการทำงานของการรับเข้า ส่งออก
/SUMMARY	สรุปสถิติเป็นรายงาน

รายงานที่แสดงประกอบด้วย

ลำดับชั้นการทำงานของระบบปฏิบัติการแควคี่เอมเอสแบ่งเป็น 4 แบบวิธี ใจกลาง (Kernel) , ะกระทำกร (Executive), ตัวกำกับดูแล (Supervisor) และผู้ใช้ (User)

แบบวิธีใจกลาง (Kernel Mode) เป็นการทำงานศูนย์กลางของการดำเนินงานของวีเอมเอส โดยจัดการข้อมูลในระบบปฏิบัติการวีเอมเอสใช้ในการจัดการลำดับงาน อินพุทเอาพุท และหน่วยความจำ เป็นเปอร์เซ็นต์ของเวลาที่พื้ทุกัทั้งหมด

แบบวิธีกระทำกร (Executive Mode) ใช้สำหรับการจัดการระเบียบน (Record Management Service หรือ RMS) โดยดูแลเกี่ยวกับแฟ้มบนจานแม่เหล็กทั้งหมด เป็นเปอร์เซ็นต์ของเวลาที่พื้ทุกัทั้งหมด

แบบวิธีตัวกำกับดูแล (Supervisor) เป็นดำเนินงานคำสั่ง ทั้งการเรียกใช้ อาเอมเอส และการบริการจากระบบวีเอมเอส ให้ทำงานตามที่สั่ง เป็นเปอร์เซ็นต์ของเวลาที่พื้ทุกัทั้งหมด

แบบวิธีผู้ใช้ (User Mode) ใช้ดำเนินงานโปรแกรมอรรถประโยชน์ ตัวแปล บรรณาธิกรณั และผลิตภัณัที่เพิ่มประสิทธิภาพต่างๆ เป็นเปอร์เซ็นต์ของเวลาที่พื้ทุกัทั้งหมด

เวลาเดินเครื่องเปล่า (Idle Time) เป็นเปอร์เซ็นต์ของเวลาที่พื้ทุกัทั้งหมด

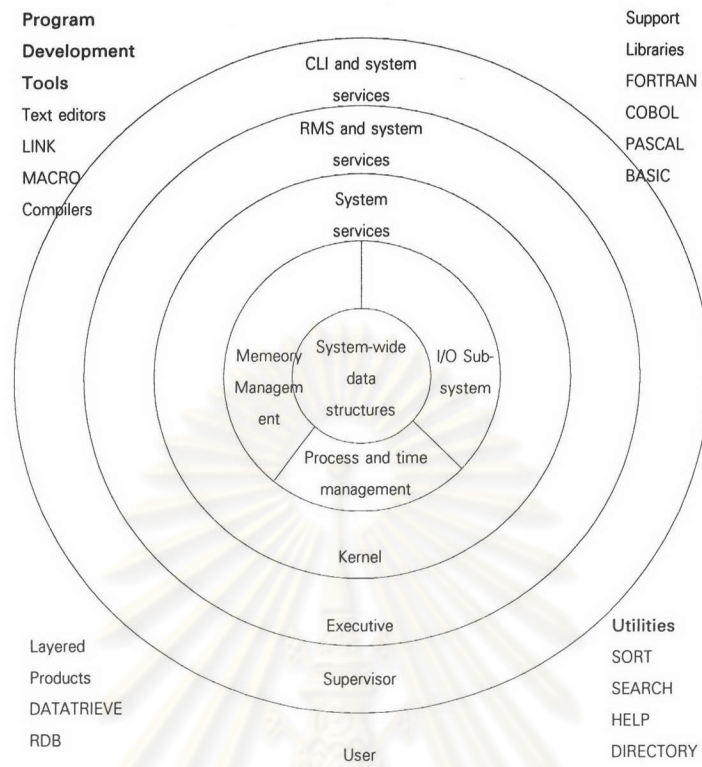
Process Count จำนวนกระบวนการที่เข้าดำเนินงานในระบบ

Page Faults จำนวนครั้งของการเกิดการบกพร่องหน้า ทั้งการบกพร่องที่จงหน่วยความจำ 512 ไบต์ (Soft Page Faults) และการบกพร่องหน้าที่ถูกสับค่า (Swap) ไปเก็บไว้ที่จานแม่เหล็กชั่วคราว เนื่องจากหน่วยความจำไม่พอ (Hard Page Faults)

Page IO Reads จำนวนครั้งของการอ่านหน้าที่ถูกสับค่าไปเก็บไว้ที่จานแม่เหล็ก (Hard Page Faults)

Buffer IO แสดงจำนวนครั้งที่ใช้อุปกรณ์ที่ต้องใช้ผ่านบัฟเฟอร์ เช่นเทอร์มัล หรือเครื่องพิมพ์

Direct IO แสดงจำนวนครั้งที่ใช้อุปกรณ์โดยตรง เช่นจานแม่เหล็ก หรือเทปแม่เหล็ก



รูปที่ 4.37 แสดงชั้น (layer) การทำงานของระบบปฏิบัติการวีเอ็มเอส (VMS Operating System)