การปรับปรุงประสิทธิภาพของระบบที่มีฐานข้อมูลขนาดใหญ่

นางสาวสุวิมล ประทุม

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2555

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ที่ส่งผ่านทางบัณฑิตวิทยาลัย

The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)
are the thesis authors' files submitted through the Graduate School.

THE IMPROVEMENT OF PERFORMANCE FOR SYSTEM WITH LARGE DATABASE

Miss Suwimon Prathum

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science Program in Software Engineering

Department of Computer Engineering

Faculty of [Engineering]

Chulalongkorn University

Academic Year 2012

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การปรับปรุงประสิทธิภาพของระบบที่มีฐานข้อมูลขนาดใหญ่
โดย	นางสาวสุวิมล ประทุม
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

…….. คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร. บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

…… ประธานกรรมการ
(รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)

…….. อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์)

…… กรรมการ
(ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล)

…… กรรมการภายนอกมหาวิทยาลัย
(ดร.อมฤต เหล่ารักพงษ์)

สุวิมล ประทุม : การปรับปรุงประสิทธิภาพของระบบที่มีฐานข้อมูลขนาดใหญ่. (THE IMPROVEMENT OF PERFORMANCE FOR SYSTEM WITH LARGE DATABASE) อ. ที่ปรึกษาวิทยานิพนธ์หลัก : รศ.ดร.วันชัย ริ้วไพบูลย์, 113 หน้า.

ระบบที่มีฐานข้อมูลขนาดใหญ่มีความจำเป็นจะต้องมีการออกแบบฐานข้อมูลที่เหมาะสม ผู้ออกแบบต้องพิจารณาถึงหลักการและแนวทางออกแบบเพื่อให้การทำงานของระบบมีประสิทธิภาพ ทั้งนี้ต้องศึกษาถึงความสัมพันธ์ของข้อมูล โครงสร้างของข้อมูล การเข้าถึงข้อมูล และกระบวนการที่โปรแกรมประยุกต์จะเรียกใช้ฐานข้อมูล ดังนั้นงานวิจัยนี้ได้นำเสนอวิธีการปรับปรุงประสิทธิภาพของระบบที่มีฐานข้อมูลขนาดใหญ่ และเสนอแนะแนวทางการปรับแต่งและออกแบบฐานข้อมูลที่เหมาะสมของระบบที่ใช้ฐานข้อมูลของกรณีศึกษา ซึ่งได้ทดสอบกับระบบรับบุคคลเข้าศึกษาในสถาบันอุดมศึกษา ประกอบด้วยระบบย่อย 3 ระบบ คือ ระบบการรับสมัครบุคคลเข้าศึกษาในสถาบันอุดมศึกษา ระบบการรวมคะแนนและจัดลำดับการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา และระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา โดยใช้ฐานข้อมูลออราเคิลและยึดหลักการปรับปรุงประสิทธิภาพของฐานข้อมูลตามแนวทางที่นำเสนอ 2 แนวทาง คือ การสร้างดรรชนีที่เหมาะสม และการปรับปรุงคำสั่งเอสคิวแอลโดยใช้พาราเลลฮินต์ ผลปรากฏว่าแนวทางที่นำเสนอนั้นสามารถช่วยให้ประสิทธิภาพของระบบดีขึ้น ได้แก่ ระบบการรับสมัครบุคคลเข้าศึกษาในสถาบันอุดมศึกษาดีขึ้นร้อยละ 44.42 ระบบการรวมคะแนนและจัดลำดับการคัดเลือกดีขึ้นร้อยละ 51.12 และระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษาดีขึ้นร้อยละ 33.89

ภาควิชา...วิศวกรรมคอมพิวเตอร์..... ลายมือนิสิต.....
สาขาวิชา...วิศวกรรมซอฟต์แวร์..... ลายมือ อ.ที่ปรึกษาวิทยานิพนธ์หลัก.....
ปีการศึกษา....2555.....

5270564721 : MAJOR SOFTWARE ENGINEERING

KEY WORD: PERFORMANCE TUNING / DATABASE / OPTIMIZATION

SUWIMON PRATHUM: THE IMPROVEMENT OF PERFORMANCE FOR SYSTEM WITH LARGE DATABASE. ADVISOR: ASSOC.PROF.WANCHAI RIVEPIBOON, Ph.D., 113 pp.

Systems with large databases, it is necessary to have a proper database design. Database designer must consider the principles and guidelines designed to ensure the effective of the system. To study the relationship of the data, structure of the data, access to information and the application access to database. This research presented a method to improve the performance of systems with large databases, design of database performance monitoring tools, with the suggestions for appropriate designing and fine-tuning. This research using the Central University Admissions System (CUAS) as the case study. The system consisted of three subsystems, which were of Online Registration, Sum Score and Selection Process, and The Online Announcement. This research using Oracle database by using design principles to guide the design of a relational database by constructing the suitable index and apply the SQL commands with the Parallel Hints. Results of this research, the database performance monitoring tool was developed, And the performance of the database design as compared to the old database from the case study: The improvement of Subsystem1 was better than 44.42%, Subsystem2 was better than 51.12% and Subsystem3 was better than 33.89%.

Department :Computer Engineering Student's Signature……

Field of Study :Software Engineering Advisor's Signature……..

Academic year :2012

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลือจากรองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ของข้าพเจ้า ขอกราบขอบพระคุณที่ได้ให้คำแนะนำและข้อเสนอแนะต่างๆ และคอยห่วงใยให้กำลังใจ ตลอดระยะเวลาในการทำวิทยานิพนธ์

ขอกราบขอบพระคุณรองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ ซึ่งเป็นประธานกรรมการสอบวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล และ ดร.อมฤต เหล่ารักพงษ์ ซึ่งเป็นกรรมการสอบวิทยานิพนธ์ ที่ได้สละเวลาและให้คำแนะนำต่างๆ ที่เป็นประโยชน์อย่างยิ่งต่อการจัดทำวิทยานิพนธ์ฉบับนี้และขอกราบขอบพระคุณอาจารย์ทุกท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับผู้เขียน รวมถึงคำปรึกษา และข้อชี้แนะต่างๆ ตลอดระยะเวลาที่ผู้เขียนได้ศึกษาเล่าเรียนในระดับปริญญามหาบัณฑิตนี้

ขอขอบคุณสมาคมอธิการบดีแห่งประเทศไทยที่ได้เอื้อประโยชน์ในการใช้เครื่องแม่ข่ายเพื่อทดสอบระบบในวิทยานิพนธ์นี้ และขอกราบขอบพระคุณรองศาสตราจารย์สมชาย ทยานยง และบิดามารดาที่คอยให้กำลังใจ ห่วงใย และสนับสนุนผู้เขียนเสมอมา

ท้ายที่สุดขอขอบคุณพี่ๆ เพื่อนๆ และน้องๆ ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัยที่คอยให้คำแนะนำ และให้กำลังใจตลอดการจัดทำวิทยานิพนธ์

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... ง

บทคัดย่อภาษาอังกฤษ..... จ

กิตติกรรมประกาศ..... ฉ

สารบัญ..... ช

สารบัญตาราง..... ญ

สารบัญภาพ..... ฏ

บทที่ 1 บทนำ..... 1

1.1 ที่มาและความสำคัญของงานวิจัย..... 1

1.2 วัตถุประสงค์ของงานวิจัย..... 2

1.3 ขอบเขตของงานวิจัย..... 3

1.4 ขั้นตอนและวิธีการวิจัย..... 3

1.5 ประโยชน์ที่ได้รับ..... 3

1.6 ผลงานตีพิมพ์..... 3

บทที่ 2 ทฤษฎี และงานวิจัยที่เกี่ยวข้อง..... 5

2.1 ทฤษฎีที่เกี่ยวข้อง..... 5

2.1.1 นิยามฐานข้อมูลขนาดใหญ่..... 5

2.1.2 การปรับแต่งประสิทธิภาพของฐานข้อมูลออราเคิล..... 5

2.1.3 การปรับแต่งฐานข้อมูลของฐานข้อมูลเชิงสัมพันธ์..... 6

2.1.4 แนวทางการออกแบบฐานข้อมูลเชิงสัมพันธ์..... 8

2.1.5 ภาษาเอสคิวแอล..... 13

2.1.6 ออราเคิล..... 14

2.2 งานวิจัยที่เกี่ยวข้อง..... 22

2.2.1 งานวิจัยเรื่อง Optimization of Bulk Operation Performances Within Oracle Database..... 22

2.2.2 งานวิจัยเรื่อง Analysis of the Query Plan for Assessing

หน้า

Optimizer Performance of SQL Select Statement…… 23
2.2.3 งานวิจัยเรื่อง Non-linear Optimization of Performance Functions for Autonomic Database Performance Tuning…... 23
2.2.4 งานวิจัยเรื่อง Self-Tuning for SQL Performance in Oracle Database 11g…….. 23
2.2.5 งานวิจัยเรื่อง Knowledge Management in Autonomic Database Performance Tuning…….. 24
2.2.6 Oracle Enterprise Manager…….. 25
บทที่ 3 แนวคิดและขั้นตอนการวิจัย…….. 26
3.1 ขั้นตอนการดำเนินการวิจัย…….. 26
3.2 โครงสร้างฐานข้อมูลของระบบ…….. 29
3.3 การออกแบบและปรับปรุงประสิทธิภาพของฐานข้อมูล…… 35
3.4 การพัฒนาเครื่องมือเพื่อวัดประสิทธิภาพ…… 48
3.5 การทดสอบการปรับปรุงฐานข้อมูล…… 48
3.6 การประเมินผลการปรับปรุงประสิทธิภาพ…….. 51
บทที่ 4 การออกแบบและพัฒนาเครื่องมือ…… 52
4.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ…… 52
4.2 การออกแบบและโครงสร้างของเครื่องมือ…….. 52
4.3 โครงสร้างฐานข้อมูลของเครื่องมือ……. 55
บทที่ 5 การทดสอบและประเมินผล…….. 57
5.1 กำหนดตัววัดประสิทธิภาพ…….. 57
5.2 ผลการปรับปรุงประสิทธิภาพของระบบกรณีศึกษา…….. 57
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ…….. 64
6.1 สรุปผลการวิจัย…… 64
6.2 ข้อเสนอแนะ…… 64
รายการอ้างอิง..... 66

หน้า

ภาคผนวก..... 67
ประวัติผู้เขียนวิทยานิพนธ์..... 113

สารบัญตาราง

ตารางที่		หน้า
3.1	ผลการประเมินประสิทธิภาพการไม่ใช้ดรรชนี……	39
3.2	ผลการประเมินประสิทธิภาพการใช้ดรรชนีแบบปกติ……	39
3.3	ผลการประเมินประสิทธิภาพการใช้ดรรชนี Bitmap ……	40
3.4	ผลการประเมินประสิทธิภาพการใช้ดรรชนีแบบ Reverse……	40
3.5	เปรียบเทียบการใช้ Parallel Hint และไม่ใช้ Parallel Hint……	44
5.1	ผลการปรับปรุงประสิทธิภาพของระบบรับสมัครคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา……..	57
5.2	แสดงคำสั่งเอสคิวแอลการปรับปรุงระบบรับสมัครคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา……..	58
5.3	ผลการปรับปรุงประสิทธิภาพของระบบรวมคะแนนและจัดลำดับการคัดเลือก……..	60
5.4	แสดงคำสั่งเอสคิวแอลการปรับปรุงระบบรวมคะแนนและจัดลำดับการคัดเลือก..	60
5.5	ผลการปรับปรุงประสิทธิภาพของระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา……..	62
5.6	แสดงคำสั่งเอสคิวแอลการปรับปรุงระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา……	63
ก-1	พจนานุกรมข้อมูลตาราง ONET_51……..	70
ก-2	พจนานุกรมข้อมูลตาราง ONET_52……..	71
ก-3	พจนานุกรมข้อมูลตาราง ONET_53……..	71
ก-4	พจนานุกรมข้อมูลตาราง ONET_54……..	72
ก-5	พจนานุกรมข้อมูลตาราง GATPAT_353……...	73
ก-6	พจนานุกรมข้อมูลตาราง GATPAT_154……...	74
ก-7	พจนานุกรมข้อมูลตาราง GATPAT_155……...	75
ก-8	พจนานุกรมข้อมูลตาราง GATPAT_255……...	75
ก-9	พจนานุกรมข้อมูลตาราง register……	77
ก-10	พจนานุกรมข้อมูลตาราง admhistory……...	80

ตารางที่		หน้า
ก-11	พจนานุกรมข้อมูลตาราง school.....	82
ก-12	พจนานุกรมข้อมูลตาราง factname.....	83
ก-13	พจนานุกรมข้อมูลตาราง DBNM.....	83
ก-14	พจนานุกรมข้อมูลตาราง DBSUBJ.....	84
ก-15	พจนานุกรมข้อมูลตาราง DBGRP.....	85
ก-16	พจนานุกรมข้อมูลตาราง DBGRD.....	86
ก-17	พจนานุกรมข้อมูลตาราง DBSUBJ_DEL.....	88
ก-18	พจนานุกรมข้อมูลตาราง DBGRP_DEL.....	88
ก-19	พจนานุกรมข้อมูลตาราง SPT.....	88
ก-20	พจนานุกรมข้อมูลตาราง result.....	89
ก-21	พจนานุกรมข้อมูลตาราง intview.....	90
ก-22	พจนานุกรมข้อมูลตาราง v$sql_monitor.....	90
ก-23	พจนานุกรมข้อมูลตาราง wrh$_sqltext.....	93
ก-24	พจนานุกรมข้อมูลตาราง wrh$_sql_plan.....	95
ก-25	พจนานุกรมข้อมูลตาราง wrh$_sqlstat.....	96
ก-26	พจนานุกรมข้อมูลตาราง v$sql_monitor.....	97

สารบัญภาพ

ภาพที่		หน้า
2.1	แสดงตัวอย่างสถานะของฐานข้อมูลสำหรับฐานข้อมูลเชิงสัมพันธ์.....	7
2.2	แสดงตัวอย่างนอร์มาไลเซชัน ของ 1NF……	9
2.3	แสดงนอร์มาไลเซชันของ 2NF และ 3NF……..	10
2.4	สถาปัตยกรรมของออราเคิล……..	14
3.1	แสดงภาพขั้นตอนการดำเนินงานวิจัย……..	24
3.2	แสดงภาพรวมวิธีการวิจัย……..	25
3.3	แสดงโครงสร้างของระบบที่ใช้งานในปัจจุบัน……...	26
3.4	แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลของระบบรับสมัคร……	28
3.5	แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลของระบบรวมคะแนนและจัดลำดับการคัดเลือก……..	29
3.6	แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลของระบบประกาศผล……...	30
3.7	แสดงรายการครรชนีบน Enterprise Manager ของระบบปัจจุบัน……	35
3.8	แสดงรายการครรชนีที่ใช้ระบบปัจจุบัน.....……	37
3.9	กราฟเปรียบเทียบการใช้ดรรชนี.....……...	41
3.10	แสดงแผนการสืบค้นด้วยการใช้ Parallel Hint ……..	42
3.11	แสดงแผนการสืบค้นโดบไม่ใช้ Parallel Hint ……	42
3.12	แผนภาพการไหลของข้อมูลสำหรับการ Join ตาราง……	43
3.13	กราฟเปรียบเทียบการใช้และไม่ใช้ Parallel Hint ……...	47
3.14	กราฟเปรียบเทียบประสิทธิภาพของระดับของ Parallel Hint ……..……	47
3.15	แสดงการทำงานของอ๊อพติไมเซอร์ในการประมวลผลคำสั่ง……	49
3.16	ตัวอย่างแผนการสืบค้นของคำสั่ง Select ก่อนปรับปรุง……...	50
3.17	ตัวอย่างแผนการสืบค้นของคำสั่ง Select หลังปรับปรุง.....	50
4.1	โครงสร้างการทำงานของเครื่องมือ……...	53
4.2	ตัวอย่างขั้นตอนการประมวลผลคำสั่ง Select……..	53
4.3	แสดงหน้าจอการแสดงผลประสิทธิภาพของคำสั่งเอสคิวเอสในระบบ……	54
4.4	แสดงหน้าจอการแสดงผลรายละเอียดของแผนการสืบค้น……..	55

ภาพที่ หน้า

4.5 โครงสร้างฐานข้อมูลของเครื่องมือ..... 56

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ระเบียบวิธีการทางวิศวกรรมซอฟต์แวร์ (Software engineering methodology) [1] จะระบุรายละเอียดเกี่ยวกับโครงสร้างของฐานข้อมูลที่โปรแกรมจะใช้และเข้าถึงฐานข้อมูล ส่วนระเบียบวิธีการออกแบบฐานข้อมูล (Database design methodology) นั้น จะกล่าวถึงแนวคิดในการกำหนดวิธีการดำเนินการ (Operation) กับวัตถุฐานข้อมูล ซึ่งทั้งสองส่วนมีความเกี่ยวเนื่องกันอย่างมาก งานส่วนหนึ่งในการจัดทำฐานข้อมูลจะต้องประกอบด้วยการออกแบบ การพัฒนา และการทดสอบแอปพลิเคชัน (Application) โดยที่กระบวนการทำงานด้านต่างๆ ย่อมต้องการประสิทธิภาพและความรวดเร็วในการทำงาน ระบบแอปพลิเคชันที่เกี่ยวข้องกับฐานข้อมูลเช่นเดียวกันที่มีความสำคัญและต้องการความรวดเร็ว ผู้ดูแลฐานข้อมูลมักพบปัญหาด้านสมรรถนะ (Performance) บ่อยครั้ง ดังนั้นควรจะมีวิธีการแก้ไขปรับปรุงอย่างไรให้ดีขึ้น ซึ่งการปรับแต่งฐานข้อมูล (Database Tuning) [2] เป็นอีกวิธีหนึ่งที่ช่วยแก้ปัญหาความล่าช้าของฐานข้อมูล การปรับแต่งฐานข้อมูลนั้นมี 2 แนวทาง คือ การปรับด้านฮาร์ดแวร์ (Hardware) และด้านซอฟต์แวร์(Software) สำหรับการปรับด้านฮาร์ดแวร์มีข้อดีคือ ง่ายและไม่ยุ่งยาก แต่ข้อเสียคือค่าใช้จ่ายมากกว่าการปรับด้านซอฟต์แวร์ ดังนั้นเพื่อลดค่าใช้จ่ายจึงควรปรับแต่งด้านซอฟต์แวร์ ดังที่ออราเคิล (Oracle) เคยนำเสนอไว้ว่าความสำคัญของสิ่งแวดล้อมต่างๆ ที่มีผลต่อประสิทธิภาพของฐานข้อมูลจะเห็นได้ว่าหน่วยประมวลผลกลาง (CPU) มีผลต่อประสิทธิภาพของฐานข้อมูลคิดเป็นร้อยละ 9 , Storage I/O ร้อยละ 43 ส่วนความซับซ้อนอื่นๆ ได้แก่ เอสคิวแอล (SQL), ฐานข้อมูล และการออกแบบแอปพลิเคชัน (Application Design) มีผลกระทบถึงร้อยละ 48 [3] ซึ่งจะว่าเห็นความสำคัญของส่วนแอสคิวแอล ฐานข้อมูล และการออกแบบแอปพลิเคชันมีร้อยละของผลกระทบมากที่สุด ผู้วิจัยเห็นความจำเป็นที่ควรให้ความสำคัญในการปรับแต่งประสิทธิภาพในส่วนนี้เพื่อให้เกิดประโยชน์สูงสุดต่อระบบงาน

ปัญหาที่พบสำหรับระบบที่ใช้ฐานข้อมูลคือ 1) การใช้คำสั่งเอสคิวแอลไม่เป็นไปตามหลักการใช้คำสั่งที่มีประสิทธิภาพสูงสุด 2) ปัญหาสิ่งสิ้นเปลืองที่ไม่จำเป็นของการบันทึกรายการ (Logging) และปัญหาการกำจัดข้อมูลที่ไม่จำเป็น 3) ปัญหาขนาดบัฟเฟอร์ (Buffer) และตารางเวลาของการประมวลผล จากปัญหาดังกล่าวผู้วิจัยเล็งเห็นความสำคัญของการสร้างวิธีการในการปรับปรุงเพื่อเพิ่มประสิทธิภาพของระบบ โดยทดสอบกับระบบการคัดเลือกบุคคลเข้าศึกษา

ในสถาบันอุดมศึกษา (CUAS : Central University Admissions System) [4] ที่มีฐานข้อมูลขนาดมากกว่า 10 กิกะไบต์ (Gigabyte) เป็นระบบการรับบุคคลที่จบระดับชั้นมัธยมศึกษาตอนปลาย หรือเทียบเท่าเข้าศึกษาในสถาบันอุดมศึกษา ทั้งสถาบันของรัฐบาล และเอกชน ปัจจุบันดำเนินการโดยใช้ข้อมูลสำหรับการคัดเลือก ได้แก่ ข้อมูลการทดสอบทางการศึกษาแห่งชาติขั้นพื้นฐาน หรือ โอเน็ต (O-NET : Ordinary National Education Test), ความถนัดทั่วไป (GAT) และความถนัดทางวิชาชีพและวิชาการ (PAT) ซึ่งกระบวนการประมวลผลนั้นจะนำคะแนนจากผลการทดสอบเหล่านี้มาใช้ประมวลผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา เนื่องจากข้อมูลที่มีอยู่เป็นจำนวนมากจึงจำเป็นต้องมีการจัดการข้อมูลที่เหมาะสม รวมทั้งการเรียกใช้ข้อมูลต้องมีความรวดเร็วเพื่อตอบสนองต่อผู้ต้องการค้นหาข้อมูลที่มีอยู่จำนวนมาก เพราะฉะนั้นผู้ดูแลฐานข้อมูลควรมีวิธีการเพื่อทำให้ประสิทธิภาพของระบบดีขึ้นและสิ้นเปลืองทรัพยากรน้อยที่สุด

ดังนั้น งานวิจัยนี้จะนำเสนอวิธีการปรับแต่งฐานข้อมูลให้เหมาะสมเพื่อเพิ่มประสิทธิภาพของระบบที่ใช้ฐานข้อมูลด้วยวิธีการปรับแต่งและเปลี่ยนแปลงความสัมพันธ์ของโครงสร้างฐานข้อมูลที่มีอยู่ให้เป็นไปตามหลักทฤษฎีการออกแบบฐานข้อมูลเชิงสัมพันธ์ มุ่งเน้นการวิเคราะห์และสร้างวิธีการสำหรับการปรับปรุงประสิทธิภาพของระบบที่มีฐานข้อมูลขนาดใหญ่ และเพื่อเสนอแนะแนวทางในการออกแบบฐานข้อมูลที่เหมาะสม โดยใช้ระบบการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษาเป็นกรณีศึกษา

1.2 วัตถุประสงค์ของงานวิจัย

1.2.1 เพื่อวิเคราะห์และสร้างวิธีการสำหรับการปรับปรุงประสิทธิภาพของระบบที่มีฐานข้อมูลขนาดใหญ่

1.2.2 เพื่อเสนอแนะแนวทางในการออกแบบฐานข้อมูลที่เหมาะสม

1.3 ขอบเขตการวิจัย

1.3.1 การศึกษาวิทยานิพนธ์นี้จะใช้ฐานข้อมูลในระบบการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษาเป็นกรณีศึกษา ซึ่งประกอบด้วยระบบย่อย ดังนี้

1) ระบบการรับสมัครบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

2) ระบบการรวมคะแนนและจัดลำดับการคัดเลือก (Selection)

3) ระบบการประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

1.3.2 งานวิจัยนี้จะใช้ฐานข้อมูลออราเคิลเวอร์ชัน 11 จี ในการทดสอบ

1.3.3 การปรับแต่งจะยึดตามหลักทฤษฎีการออกแบบฐานข้อมูลเชิงสัมพันธ์

1.3.4 งานวิจัยนี้จะเปรียบเทียบประสิทธิภาพของโปรแกรมประยุกต์ของระบบหลังการปรับแต่งฐานข้อมูล

1.3.5 ซอฟต์แวร์สำหรับทดสอบประสิทธิภาพจะให้ผลลัพธ์ในรูปแบบเวลาที่ใช้การประมวลผลของโปรแกรมประยุกต์ และลักษณะของการออกแบบฐานข้อมูล

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ได้วิธีการปรับแต่งฐานข้อมูลที่เหมาะสมและมีประสิทธิภาพดีขึ้น

1.4.2 สามารถช่วยเสนอแนะแนวทางสำหรับการออกแบบฐานข้อมูลที่เหมาะสมให้กับนักพัฒนาระบบ

1.4.3 สามารถนำวิธีการปรับแต่งฐานข้อมูลไปปรับใช้กับระบบที่ใช้ฐานข้อมูลอื่นๆ ได้

1.4.4 ได้เครื่องมือแสดงผลประสิทธิภาพของฐานข้อมูลที่เหมาะสมกับงาน

1.5 วิธีการดำเนินการวิจัย

1.5.1 ศึกษาโมเดลและโครงสร้างของระบบการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

1.5.2 ศึกษาการปรับปรุงประสิทธิภาพของฐานข้อมูล

1.5.3 ศึกษาการออกแบบฐานข้อมูลเชิงสัมพันธ์

1.5.4 ออกแบบและสร้างแนวทางในการปรับปรุงประสิทธิภาพของฐานข้อมูล

1.5.5 ออกแบบและพัฒนาเครื่องมือสำหรับแสดงผลประสิทธิภาพของการปรับแต่งฐานข้อมูล

1.5.6 ทดสอบประสิทธิภาพหลังปรับแต่งฐานข้อมูล

1.5.7 ทดสอบการทำงานของเครื่องมือ

1.5.8 ทดลองและสรุปผลการวิจัย

1.5.9 จัดทำรายงานวิทยานิพนธ์

1.6 ผลงานตีพิมพ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้รับการตีพิมพ์เป็นบทความทางวิชาการในหัวข้อเรื่อง “การปรับปรุงประสิทธิภาพของฐานข้อมูลออราเคิล กรณีศึกษา : ระบบรับบุคคลเข้าศึกษาในสถาบันอุดมศึกษา” โดย สุวิมล ประทุม และ รศ.ดร.วันชัย ริ้วไพบูลย์ ในงานประชุมวิชาการ การ

ประชุมมหานครวิชาการด้านงานวิจัยและเทคโนโลยีเพื่อการศึกษา ครั้งที่ 6 จัดขึ้นในวันที่ 17 พฤษภาคม 2556 ณ คณะวิทยาการและเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีมหานคร

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

การสร้างวิธีการปรับปรุงประสิทธิภาพของของระบบที่ใช้ฐานข้อมูลจำเป็นต้องอาศัยทฤษฎีหรือกฎต่างๆ ในการรองรับวิธีการดังกล่าว ดังนั้นในงานวิจัยนี้ได้อาศัยทฤษฎีที่เกี่ยวข้องเพื่อใช้ในกระบวนการวิเคราะห์และสร้างวิธีการเพื่อการปรับปรุงประสิทธิภาพของฐานข้อมูลดังนี้

2.1.1 นิยามฐานข้อมูลขนาดใหญ่ (Large Database)

ฐานข้อมูลขนาดใหญ่ (Large Database) [3] คือ ฐานข้อมูลที่มีข้อมูลจำนวนมาก หรือระบบที่มีพื้นที่เก็บข้อมูลขนาดใหญ่ มีขนาดข้อมูลตั้งแต่ 1 กิกะไบต์ ไปถึง 1 เทราไบต์ (Terabyte)

ฐานข้อมูลขนาดใหญ่มาก (Very Large Database) คือ ฐานข้อมูลที่มีข้อมูลจำนวนมาก หรือระบบที่มีพื้นที่เก็บข้อมูลขนาดใหญ่ มีขนาดข้อมูลมากกว่า 1 เทราไบต์ หรือมีจำนวนหลายพันล้านแถว

2.1.2 การปรับแต่งประสิทธิภาพของฐานข้อมูลออราเคิล (Oracle Performance Tuning)

สำหรับการปรับแต่งประสิทธิภาพของฐานข้อมูลออราเคิล[5] มีวิธีการปรับแต่งในลักษณะจากบนลงล่าง (Top-down) ซึ่งจะเริ่มต้นจากขั้นตอนที่กว้างมากก่อนแล้วพิจารณาเฉพาะจุดในขั้นตอนถัดๆ ไป การทำการปรับแต่งประสิทธิภาพของฐานข้อมูลออราเคิลค่อนข้างซับซ้อนและสามารถทำการปรับแต่งได้ในหลายขั้นตอน ซึ่งในแต่ละขั้นตอนจะพิจารณาเฉพาะส่วนตามเป้าหมายของขั้นตอนนั้นๆ ทั้งนี้สามารถแบ่งขั้นตอนในการทำการปรับแต่งประสิทธิภาพของออราเคิลได้เป็น 4 ขั้นตอน ดังนี้

2.1.2.1 การปรับแต่งเครื่องบริการและเครือข่าย (Server & Network Tuning) ขั้นตอนนี้จะเป็นขั้นตอนแรกในการปรับแต่งเสมอ เป็นการปรับแต่งที่ง่าย และกระทบกับผู้ใช้งานและผู้พัฒนาระบบน้อยที่สุด การปรับแต่งในขั้นตอนนี้จะเกี่ยวข้องกับฮาร์ดแวร์ เช่น การเพิ่ม RAM เป็นต้น และมักกระทำการโดยผู้ดูแลระบบ (System Administrator) อย่างไรก็ดี การปรับแต่งในขั้นนี้ ไม่สามารถช่วยได้สำหรับเครื่องบริการที่มีสภาพแวดล้อม (Environment) ที่ไม่ดีอยู่แล้ว

2.1.2.2 การปรับแต่งพารามิเตอร์ของฐานข้อมูล (Instance Tuning) เป็นขั้นตอนที่อาศัยข้อมูลจากสถิติต่าง ๆ ที่ฐานข้อมูลได้มีการเก็บข้อมูลไว้ เป็นการทำที่ระดับฐานข้อมูล ซึ่งการปรับแต่งในส่วนนี้จะไม่กระทบกับนักพัฒนาระบบ และมักกระทำการโดยผู้ดูแลฐานข้อมูล พารามิเตอร์ทั้งหมดจะถูกตรวจสอบเพื่อให้มั่นใจว่าฐานข้อมูลได้ถูกกำหนดค่าอย่างเหมาะสมสำหรับการใช้งานแล้ว ในบางกรณีฐานข้อมูลอาจมีการใช้งานใน 2 รูปแบบ คือ ออนไลน์ (Online) และแบตช์ (Batch) ดังนั้น ค่าพารามิเตอร์ต่างๆ จะถูกปรับแต่งเพื่อให้เกิดการใช้งานที่เหมาะสมในขั้นตอนนี้

2.1.2.3 การปรับแต่งวัตถุ (Object Tuning) ขั้นตอนนี้เป็นการปรับแต่ง โดยพิจารณาจากการตั้งค่าต่าง ๆ ของวัตถุ เช่น การตั้งค่า PCTFREE, PCTUSED และ FREELISTS ในตาราง (Table) และดรรชนี (Index) สามารถส่งผลกระทบต่อประสิทธิภาพการทำงานของฐานข้อมูลออราเคิลด้วย การปรับแต่งในขั้นตอนนี้จะไม่กระทบกับนักพัฒนาระบบ และมักกระทำการโดยผู้ดูแลระบบฐานข้อมูล

2.1.2.4 การปรับแต่งเอสคิวแอล (SQL Tuning) ขั้นตอนนี้เป็นขั้นตอนสุดท้ายในการทำการปรับแต่ง และเป็นขั้นตอนที่ใช้เวลามากที่สุดในการทำการปรับแต่ง เนื่องจากคำสั่งเอสคิวแอล (SQL Statements) ที่เข้ามาใช้งานระบบฐานข้อมูลนั้นมีจำนวนมาก ทำให้เราไม่สามารถทราบได้ว่าคำสั่งใดที่ต้องการการปรับแต่ง ซึ่งถ้าเรามีการปรับแต่งระบบเพื่อเพิ่มประสิทธิภาพการทำงานตามขั้นตอนที่ 2.1.2.2 แล้วนั้น เราเพียงปรับแต่งประโยคที่มีประสิทธิภาพการทำงานช้ากว่าประโยคอื่นมากเท่านั้น โดยขั้นตอนนี้จะกระทบกับนักพัฒนาระบบโดยตรง เนื่องจากเป็นการปรับแต่งรูปแบบของประโยคและมักจะกระทำโดยนักพัฒนาระบบเอง

2.1.3 การปรับแต่งฐานข้อมูลของฐานข้อมูลเชิงสัมพันธ์

การปรับแต่ง (Tuning) [1] คือ กระบวนการในการแก้ไขและปรับแต่งการออกแบบฐานข้อมูลทางกายภาพอย่างต่อเนื่องโดยเฝ้าติดตามการใช้ทรัพยากร (Resource Utilization) เช่นเดียวกับกระบวนการภายในระบบจัดการฐานข้อมูล (DMBS: Database Management System) เพื่อค้นหาคอขวด (Bottleneck) เช่น การแย่งข้อมูลหรืออุปกรณ์เดียวกัน เป้าหมายคือเพื่อให้แอปพลิเคชันทำงานเร็วขึ้น เพื่อลดเวลาตอบสนองในการสืบค้น (Query) และการดำเนินการ (Transaction) และเพื่อปรับปรุงปริมาณงานที่ทำให้ช่วงเวลาหนึ่ง (Throughput) โดยรวมของการดำเนินการ

การปรับแต่งดรรชนี สำหรับเหตุผลในทำการปรับแต่งดรรชนี คือ 1) การสืบค้นอาจจะใช้เวลาในการรันนานหากไม่มีดรรชนี 2) ดรรชนีอาจจะไม่ได้ถูกเรียกใช้อย่างมีประสิทธิภาพ 3) การ

สืบค้นอาจจะทำให้เกิดสิ่งสิ้นเปลืองที่จำเป็นเป็นจำนวนมาก เนื่องจากดรรชนีอาจจะชี้ไปยังแอททริบิวต์ที่มีการเปลี่ยนแปลงบ่อย (Undergoes Frequent)

ในการปรับแต่งการออกแบบฐานข้อมูล จำเป็นต้องระบุข้อกำหนดกระบวนการเปลี่ยนอย่างต่อเนื่อง เปลี่ยนแปลงในแผนผังแนวคิดหากจำเป็น และสะท้อนการเปลี่ยนแปลงเหล่านั้นเข้าไปในสคีมาแนวความคิด (Conceptual Schema) และการออกแบบทางกายภาพ (Physical Design) สำหรับการเปลี่ยนแปลงที่เป็นไปได้ต่อการออกแบบฐานข้อมูล ได้แก่

2.1.3.1 ตารางปัจจุบันอาจรวมเข้าด้วยกันโดยการทำดีนอร์มาไลเซชัน หากมีแอททริบิวต์จากสองตารางหรือมากกว่าที่ต้องใช้พร้อมกันบ่อย และสำหรับตารางที่ได้มานั้น อาจจะมีทางเลือกในการออกแบบแบบอื่น ๆ ที่ทุกตารางนั้นอยู่ในรูปของ 3NF หรือ BCNF จึงสามารถปรับเปลี่ยนอันใดอันหนึ่ง หรือสลับตารางได้

2.1.3.2 ความสัมพันธ์ในรูป R (K, A, B, C, D, …) โดยที่ K เป็นชุดของแอททริบิวต์สำคัญ ที่อยู่ในรูปของ BCNF สามารถแบ่งเก็บอยู่ในหลายตารางที่อยู่ในรูปของ BCNF ด้วยโดยสำเนา K ที่สำคัญซ้ำลงในแต่ละตาราง ตัวอย่าง เช่น แยก R ออกเป็น ตาราง R1(K, A, B) R2(K, C, D), R3(K,…). เรียกว่า การพาร์ธิชันแนวตั้ง (Vertical Partitioning)

2.1.3.3 แอททริบิวต์จากตารางหนึ่งอาจถูกจัดเก็บในตารางอื่นได้ ถึงจะทำให้เกิดปัญหาการซ้ำซ้อน (Redundancy) หรือเกิดความผิดปกติ (Anomalies)

สิ่งที่แสดงถึงว่าต้องการปรับแต่งการสืบค้น คือ

1) การสืบค้นส่งผลให้มีการเรียกใช้ หรือ เข้าถึง (Access) ดิสก์มากเกินไป

2) แผนการการสืบค้นแสดงถึงดรรชนีที่เกี่ยวข้องที่ไม่ได้ถูกใช้

3) การสืบค้นที่มีประสิทธิภาพหลาย ๆ ตัวไม่ได้ใช้ดรรชนี หากมีสมการทางคณิตศาสตร์ การเปรียบเทียบเชิงตัวเลขของแอททริบิวต์ที่ต่างขนาดกันหรือการเปรียบเทียบค่าว่างและการเปรียบเทียบซับสตริงอยู่ด้วย

4) ดรรชนีที่มักจะไม่ค่อยถูกใช้สำหรับคำสั่งที่มีความซับซ้อนที่ใช้คำสั่ง IN

5) ในการใช้คำสั่ง Distinct นั้น อาจเกิดปัญหาความซ้ำซ้อน (Redundancy) ซึ่งเป็นสิ่งที่เราควรเลี่ยง โดยที่ผลลัพธ์จะไม่เปลี่ยนแปลง

6) ควรหลีกเลี่ยงการใช้งานตารางผลชั่วคราว (Temporary Result Table) ที่ไม่จำเป็น โดยการยุบรวมการสืบค้นเข้าด้วยกันถึง ยกเว้นแต่ว่าความสัมพันธ์แบบชั่วคราวนั้นๆ จำเป็นสำหรับการประมวลผลต่อ

7) หากเป็นไปได้ว่ามีหลายทางเลือกในการ Join

8) ให้เลือกใช้การจัดกลุ่มดรรชนีและหลีกเลี่ยงการเปรียบเทียบสตริง (String)

9) ลำดับของตารางจากคำสั่ง From อาจจะกระทบต่อกระบวนการ Join

10) การสืบค้นที่มีประสิทธิภาพบางตัว เมื่อทำงานในลักษณะที่มีคำสั่งความซับซ้อนนั้นอาจมีประสิทธิภาพด้อยกว่าการใช้คำสั่งที่ไม่มีความซับซ้อน

11) มีแอปพลิเคชันจำนวนมากที่ใช้การสร้างมุมมอง เพื่อกำหนดขอบเขตของข้อมูลที่สนใจสำหรับแอปพลิเคชันนั้นๆ แต่บางครั้งมุมมองเหล่านี้กลายเป็นผลไม่ดี (Overkill) ตัวเอง

12) การสืบค้นด้วยเงื่อนไขการเลือกหลายทางเลือก (Multiple Selection) ที่เชื่อมด้วย OR อาจจะไม่เรียก การสืบค้นที่มีประสิทธิภาพขึ้นมาเพื่อใช้ดรรชนี ดังนั้นการสืบค้นแบบนั้นควรถูกแบ่งออกมา และ แสดงในรูปแบบ Union การสืบค้น โดยแต่ละตัวจะมีเงื่อนไขในแอทตริบิวต์ที่จะทำนำไปสู่ดรรชนีที่จะใช้งาน

2.1.4 แนวทางการออกแบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database Design)

การออกแบบฐานข้อมูลเชิงสัมพันธ์ [1] คือ การจัดกลุ่มของแอทตริบิวต์เพื่อจะทำการรวมเค้าร่างความสัมพันธ์ (Relation Schemas) ที่ดี เค้าร่างความสัมพันธ์ สามารถแบ่งออกเป็นสองระดับคือ ระดับมุมมองเชิงตรรกะของผู้ใช้งาน (The Logical User View Level) และระดับความสัมพันธ์พื้นฐานของถังข้อมูล (The Storage Base Relation Level)

เริ่มพิจารณาถึงแนวทางอย่างไม่เป็นทางการในการออกแบบเชิงสัมพันธ์ที่เหมาะสมก่อน หลังจากนั้นจึงทำการพิจารณาถึงแนวทางอย่างเป็นทางการในการเชื่อมโยงฟังก์ชัน (Functional Dependencies) และบรรทัดฐาน (Normal Forms) ได้แก่ 1NF (First Normal Form), 2NF (Second Normal Form), 3NF (Third Normal Form) และ BCNF (Boyce-Codd Normal Form)

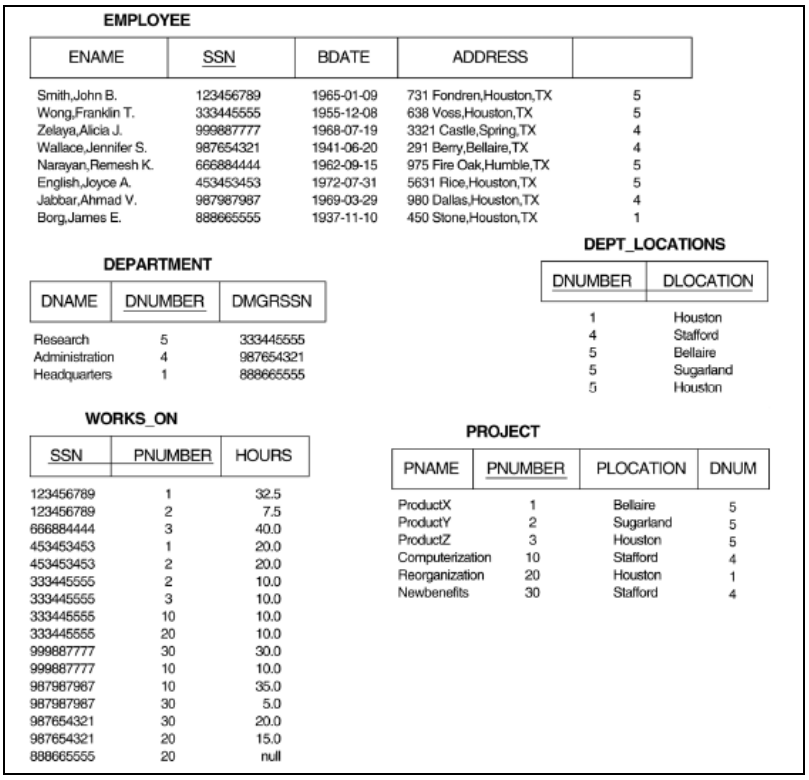
การเชื่อมโยงประเภทอื่นๆ นอกจากบรรทัดฐาน คือ อัลกอริทึมในการออกแบบเชิงสัมพันธ์โดยวิธีการสังเคราะห์ แนวทางในการออกแบบฐานข้อมูลเชิงสัมพันธ์ มีดังนี้

แนวทางที่ 1: เมื่อพิจารณาอย่างไม่เป็นทางการ จะเห็นได้ว่าในแต่ละทูเพิล (Tuple) ของรีเลชันควรจะแสดงถึงค่าเอนทิตี หรืออินสแตนซ์ของความสัมพันธ์ (Relationship Instance) ซึ่งจะนำไปประยุกต์ใช้ได้กับความสัมพันธ์เดี่ยว (Individual Relations) และแอทตริบิวต์ของรีเลชันนั้น

1) แอทตริบิวต์ของเอนทิตีที่ต่างกัน (EMPLOYEEs, DEPARTMENTs, PROJECTs) ไม่ควรที่จะถูกรวมเข้าไว้ในความสัมพันธ์เดียวกัน
2) จะนำคีย์นอก (Foreign Key) มาเพื่อใช้อ้างถึงเอนทิตีอื่น ๆ

3) ควรจะเก็บเอนทิตีและแอทตริบิวต์ความสัมพันธ์ (Relationship Attributes) แยกกันให้ได้มากที่สุดเท่าที่จะเป็นไปได้

4) การออกแบบสคีมา (Schema) หนึ่ง สามารถที่จะอธิบายด้วยความสัมพันธ์อย่างง่ายๆ โดยซีแมนติคส์ (Semantics) ของแอทตริบิวต์ ควรที่จะสื่อได้ง่ายๆ ตัวอย่างแสดงดังภาพที่ 2.1 ดังนี้



EMPLOYEE

ENAME	SSN	BDATE	ADDRESS	
Smith,John B.	123456789	1965-01-09	731 Fondren,Houston,TX	5
Wong,Franklin T.	333445555	1955-12-08	638 Voss,Houston,TX	5
Zelaya,Alicia J.	999887777	1968-07-19	3321 Castle,Spring,TX	4
Wallace,Jennifer S.	987654321	1941-06-20	291 Berry,Bellaire,TX	4
Narayan,Ramesh K.	666884444	1962-09-15	975 Fire Oak,Humble,TX	5
English,Joyce A.	453453453	1972-07-31	5631 Rice,Houston,TX	5
Jabbar,Ahmad V.	987987987	1969-03-29	980 Dallas,Houston,TX	4
Borg,James E.	888665555	1937-11-10	450 Stone,Houston,TX	1

DEPARTMENT

DNAME	DNUMBER	DMGRSSN
Research	5	333445555
Administration	4	987654321
Headquarters	1	888665555

DEPT_LOCATIONS

DNUMBER	DLOCATION
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

WORKS_ON

SSN	PNUMBER	HOURS
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	null

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

ภาพที่ 2.1 แสดงตัวอย่างสถานะของฐานข้อมูลสำหรับ Relation database schema [1]

แนวทางที่ 2: การออกแบบสคีมา (Schema) ที่ไม่ทำให้ได้รับผลกระทบจากการอัพเดทอะนอร์มาลีในการแทรก การลบ หรือการปรับปรุงโดยหากเป็นเช่นนั้น หากมีการเกิดอะนอร์มาลีอย่างใดอย่างหนึ่ง จำเป็นต้องคำนึงถึงเมื่อทำการสร้างแอปพลิเคชัน

แนวทางที่ 3: ควรออกแบบรีเลชันให้มีค่าว่าง (Null value) อยู่น้อยที่สุด แอทตริบิวต์ที่เป็นว่าง มักจะสามารถนำไปสร้างความสัมพันธ์หรือรีเลชันใหม่พร้อมกับค่าคีย์หลัก (Primary Key) ได้ เนื่องจากการมีค่าว่างอาจทำให้ไม่สามารถใช้แอทตริบิวต์ได้ หรือใช้ได้แต่ไม่ถูกต้อง อาจเกิดแอทตริบิวต์ที่มีค่าแต่ไม่ทราบค่าเกิดขึ้น และอาจมีค่าเกิดขึ้นแต่ไม่สามารถนำมาใช้งานได้

แนวทางที่ 4: ไม่ควรออกแบบรีเลชันเพื่อสนองความต้องการแค่ในส่วนของเงื่อนไข Join และการทำรวมแบบธรรมชาติ (Natural Join) ของรีเลชันใดๆ ไม่ควรสร้างทูเพิลแบบสฟิวเรียส (Spurious) ซึ่งการออกแบบที่ไม่ดีนักของฐานข้อมูลเชิงสัมพันธ์จะก่อให้เกิดผลลัพธ์ที่ผิดพลาด

สำหรับการดำเนินการ (Oparation) Join อาจจะนำคุณสมบัติที่เรียกว่า “Lossless Join” ไปใช้สำหรับยืนยันผลลัพธ์ที่มีความสำคัญสำหรับโอเปอเรชัน Join

2.1.4.1 ฟังก์ชันนัลดีเพนเดนซี (FDs : Functional Dependencies) ถูกนำไปใช้ในการกำหนดวิธีในการวัดความสมบูรณ์ของการออกแบบเชิงสัมพันธ์อย่างเป็นทางการ หรือเรียกได้ว่าเป็นการวัด “Goodness” ของการออกแบบ ฟังก์ชันนัลดีเพนเดนซีและคีย์จะถูกนำไปใช้ในการกำหนดบรรทัดฐาน (Normal Forms) นอกจากนั้นฟังก์ชันนัลดีเพนเดนซียังทำหน้าที่เป็นข้อจำกัดที่ทำการดีไรฟ์ (Derive) มาจากความหมายและความสัมพันธ์ภายใน (Interrelationships) ของแอทตริบิวต์ของข้อมูล

2.1.4.2 นอร์มาไลเซชัน (Normalization) คือ กระบวนการเปลี่ยนรีเลชันที่“ไม่ดี” หรือไม่น่าพึงพอใจโดยการแตกออกเป็นรีเลชันที่เล็กลงเพื่อให้รีเลชันมีลักษณะเป็นรีเลชันที่ดี

2.1.4.3 ดีนอร์มาไลเซชัน (Denormalization) เป็นการตัดสินใจการออกแบบเพื่อเพิ่มความเร็วในการการสืบค้น โดยมีวัตถุประสงค์หลักคือการจัดแยกแอทตริบิวต์ที่เกี่ยวข้องกันในเชิงตรรกะ (Logically related) ให้อยู่ในรูปของตาราง เพื่อลดการซ้ำซ้อน (Redundancy) และหลีกเลี่ยงการอัพเดทอะนอร์มาลี ซึ่งเป็นสาเหตุให้เกิดการทำงานที่เป็นสิ่งสิ้นเปลืองที่จำเป็น เพื่อรักษาความถูกต้องของฐานข้อมูล วัตถุประสงค์ของการทำดีนอร์มาไลเซชัน คือ การปรับปรุงประสิทธิภาพของการสืบค้นและการดำเนินการที่เกิดขึ้นเป็นประจำ (โดยทั่วไปนักออกแบบจะรวมเข้าไปในตารางแอทตริบิวต์ ที่จำเป็นต้องใช้ในการตอบการสืบค้นหรือการสร้างรายงาน ดังนั้นจึงควรหลีกเลี่ยงการรวมเข้ากับตารางอื่น) อาจมองได้ว่าการดีนอร์มาไลเซชันเป็นการเทรดออฟ (Trade Off) ระหว่างประสิทธิภาพของการอัพเดทและการทำการสืบค้น

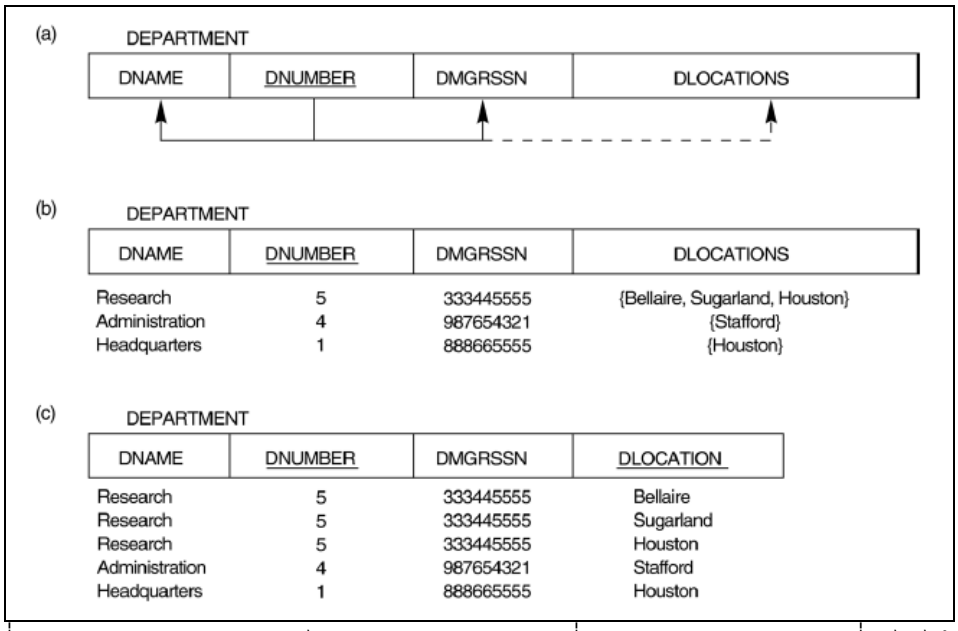
2.1.4.4 บรรทัดฐาน คือ เงื่อนไขที่ระบุว่ารีเลชันมีบรรทัดฐานสูงสุดที่ดิกรีใด หรือดิกรีของการนอร์มาไลเซชันที่รีเลชันนั้นผ่านการนอร์มัลไลซ์แล้ว ส่วนของ 2NF, 3NF, BCNF มีพื้นฐานอยู่บนคีย์ และฟังก์ชันนัลดีเพนเดนซีของเค้าร่างรีเลชันหนึ่ง ส่วน 4NF มีพื้นฐานอยู่บนคีย์และ Multi-Valued Dependencies : MVDs; 5NF มีพื้นฐานอยู่บนคีย์ และ Join Dependencies : JDs นอกจากนั้นยังต้องการส่วนของคุณสมบัติเพิ่มเติม (Additional Properties) เพื่อเป็นตัวการันตีการออกแบบความสัมพันธ์ที่ดี (Good Relation Design) การใช้งานของบรรทัดฐาน ได้แก่ 1) นอร์มาไลเซชันจะถูกนำไปใช้ในการทำงานเพื่อให้ได้ผลลัพธ์ของการออกแบบที่มีคุณภาพดี และตรงตามคุณสมบัติที่ต้องการ 2) บรรทัดฐานจะช่วยให้นักออกแบบหรือผู้ใช้ฐานข้อมูลเข้าใจข้อจำกัด (Constraints) ของฐานข้อมูลได้ชัดเจนมากยิ่งขึ้น 3) ผู้ออกแบบฐานข้อมูลไม่จำเป็นต้องนอร์มัลไลซ์ให้ถึงระดับที่สูงที่สุด(ปกติจะใช้แค่ระดับ 3NF, BCNF หรือ 4NF) 4) ดีนอร์มาไลเซชัน

(Denormalization) กระบวนการในการ Join รีเลชันที่มีบรรทัดฐานระดับสูงให้เป็นบรรทัดฐานระดับต่ำลง ซึ่งรายละเอียดของบรรทัดฐาน มีดังนี้

2.1.4.4.1 บรรทัดฐานระดับที่ 1 (1NF: First Normal Form)

1) ไม่อนุญาตรีเลชันมีแอทตริบิวต์หลายค่า (Multivalued Attribute) แอทตริบิวต์ผสม (Composite Attribute) และแอทตริบิวต์ทั้งสองแบบรวมกัน

2) ค่าของแอทตริบิวต์ที่อนุญาตสำหรับ 1NF จะมีลักษณะเป็น Single Atomic Value (หรือ Indivisible) เท่านั้น



ภาพที่ 2.2 แสดงตัวอย่างนอร์มาไลเซชัน ของ 1NF โดยที่ (a) Relation schema ที่ยังไม่ได้อยู่ใน 1NF, (b) ตัวอย่าง สถานะของความสัมพันธ์ของ DEPARTMENT, (c) 1NF เวอร์ชั่นของความสัมพันธ์เดิมพร้อมกับค่าซ้ำซ้อน[1]

2.1.4.4.2 บรรทัดฐานระดับที่ 2 (2NF: Second Normal Form) ใช้หลักของ FDs และคีย์หลัก (Primary Key)

1) แอทตริบิวต์หลัก (Prime Attribute) คือ แอทตริบิวต์ที่เป็นสมาชิกของคีย์หลัก K

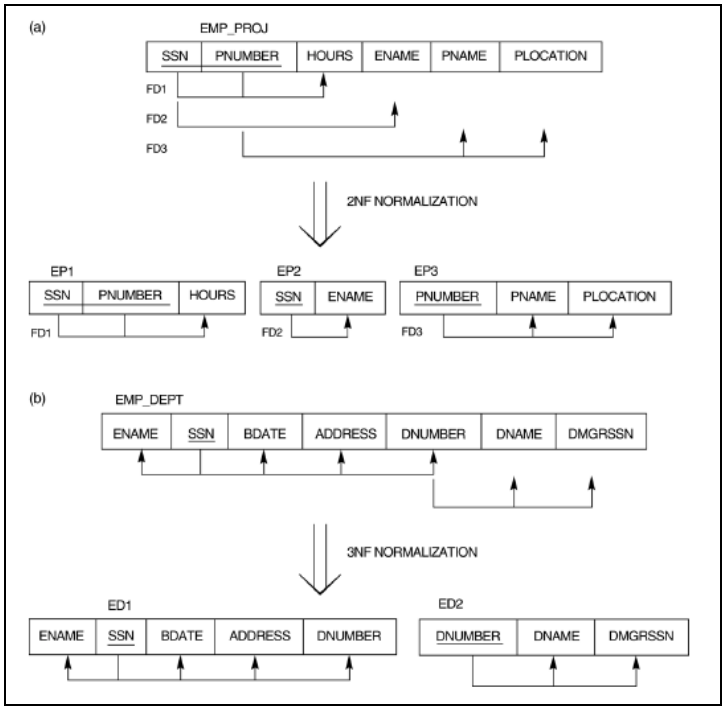
2) ฟังก์ชันนัลดีเพนเดนซีแบบสมบูรณ์ (Full Functional Dependency) คือ FD X -> Y จะเป็น FD แบบสมบูรณ์ ถ้ามีการนำแอทตริบิวต์ใดๆออกจาก X แล้วทำให้ดีเพนเดนซีนั้นถูกทำลาย ตัวอย่าง เช่น

2.1) {SSN, PNUMBER} -> HOURS เป็น FD แบบสมบูรณ์ เนื่องจากเมื่อตัด SSN หรือ PNUMBER ออกจาก {SSN, PNUMBER} จะไม่มีดี เพนเดนซี SSN -> HOURS หรือ NUMBER -> HOURS

2.2) {SSN, PNUMBER} -> ENAME เป็นฟังก์ชันนัลดีเพนเดนซีแบบบางส่วน (Partial functional dependency) เนื่องจากเมื่อตัด PNUMBERออกจาก {SSN, PNUMBER} ยังคงมีดีเพนเดนซี SSN -> ENAME เหลืออยู่

3) สำหรับเค้าร่างรีเลชัน R จะอยู่ใน 2NF ถ้าทุก Nonprime attributeใน R เป็นฟังก์ชันดีเพนเดนซีแบบเต็มที่ขึ้นอยู่กับคีย์หลัก

4) R สามารถถูกแตกย่อยออกเป็นรีเลชันแบบ 2NF ได้โดยกระบวนการนอร์มัลไลเซชัน 2NF



ภาพที่ 2.3 แสดงนอร์มาไลเซชันของ 2NF และ 3NF (a) นอร์มัลไลซ์ EMP_PROJ เข้าสู่ 2NF (b) นอร์มัลไลซ์ EMP_DEPT เข้าสู่ 3NF[1]

2.1.4.4.3 บรรทัดฐานระดับที่ 3 (3NF: Third Normal Form) Transitive functional dependency คือ FD X -> Z ที่สร้าง (derive) มาจากฟังก์ชันนัลดีเพนเดนซี X -> Y

และ Y -> Z ตัวอย่าง เช่น SSN -> DMGRSSN เป็น Transitive dependency เนื่องจากมีดีเพนเดนซี SSN -> DNUMBER และ DNUMBER -> DMGRSSN

1) เค้าร่างรีเลชัน R จะอยู่ใน 3NF ถ้า R เป็น 2NF ที่ไม่มีแอทตริบิวต์แบบ Nonprime ที่เป็น Transitive dependency ที่ขึ้นอยู่กับคีย์หลัก

2) R สามารถถูกแตกย่อยมาเป็นรีเลชัน 3NF ได้โดยกระบวนการนอมัลไลเซชัน 3NF

2.1.4.4.4 บรรทัดฐานระดับบอยส์-คอดด์ (BCNF: Boyce-Codd Normal Form)

1) เค้าร่างรีเลชัน R จะอยู่ใน Boyce-Codd Normal Form (BCNF) ถ้าฟังก์ชันนัลดีเพนเดนซี X -> A ใน R มี X เป็นซูเปอร์คีย์ของ R

2) ในแต่ละบรรทัดฐานจะถูกบังคับไว้ชัดเจน ได้แก่ ทุกรีเลชัน 2NF จะต้องอยู่ภายใน 1NF, ทุกรีเลชัน 3NF จะต้องอยู่ภายใน 2NF และทุกรีเลชัน BCNF จะต้องอยู่ภายใน 3NF

3) มีรีเลชันที่อยู่ใน 3NF แต่ไม่อยู่ใน BCNF

4) เป้าหมายคือ มีทุกๆ รีเลชันอยู่ใน BCNF (หรือ 3NF)

2.1.5 ภาษาเอสคิวแอล

ภาษาเอสคิวแอล [6] หรือภาษาสอบถามเชิงโครงสร้าง (SQL: Structured Query Language) เป็นกลุ่มของคำสั่งที่ผู้ใช้ สามารถจัดการกับฐานข้อมูลเชิงสัมพันธ์ เช่น การสร้างฐานข้อมูล การดึงข้อมูลออกจากฐานข้อมูล การเพิ่มข้อมูลลงไปในฐานข้อมูล เป็นต้น ซึ่งในปัจจุบันซอฟต์แวร์ด้านระบบจัดการฐานข้อมูลเชิงสัมพันธ์ที่รองรับภาษาเอสคิวแอล เช่น ฐานข้อมูลมายเอสคิวแอล ฐานข้อมูลออราเคิล เป็นต้น ประเภทของคำสั่งในภาษาเอสคิวแอลแบ่งออกได้เป็น 3 ระดับ คือ

2.1.5.1 ภาษาสำหรับการนิยามข้อมูล (DDL: Data Definition Language) เป็นภาษาที่ใช้ในการสร้างฐานข้อมูล เช่น ตาราง ดรรชนี วิว (View) เป็นต้น

2.1.5.2 ภาษาสำหรับการจัดการข้อมูล (DML: Data Manipulation Language) เป็นภาษาที่ใช้ในการเรียกข้อมูล หรือปรับปรุงข้อมูล

2.1.5.3 ภาษาควบคุม (DCL: Data Control Language) เป็นภาษาที่ใช้ในการกำหนดสิทธิ์ของผู้ใช้ งานในภาษาเอสคิวแอลจะต้องมีการกำหนดชนิดข้อมูลให้กับทุกๆ คุณลักษณะของรีเลชัน โดยสามารถทำการแบ่งชนิดข้อมูลในภาษาเอสคิวแอลได้ ดังนี้

1) จำนวนเลข (Numeric)

1.1) จำนวนเลขที่มีจุดทศนิยม (Decimal) เป็นประเภทข้อมูลที่มีจุดทศนิยม จะใช้ Decimal(i, d) โดยที่ i คือจำนวนตัวเลขทั้งหมด และ d คือจุดทศนิยม

1.2) จำนวนเลขที่ไม่มีจุดทศนิยม เป็นประเภทข้อมูลที่เป็นจำนวนเต็มจะใช้ integer

1.3) จำนวนจริงจะใช้ Number

2) ตัวอักขระ (Character)

2.1) ตัวอักขระแบบความยาวคงที่ (Fixed-Length Character) เป็นประเภทข้อมูลที่มีความยาวของข้อมูลคงที่จะใช้ Char(n)

2.2) ตัวอักขระแบบความยาวไม่คงที่ (Variable-Length Character) เป็นประเภทข้อมูลที่มีความยาวของข้อมูลไม่คงที่จะใช้ varchar(n)

3) วัน (Date) เป็นข้อมูลของชนิดวันที่

2.1.6 ออราเคิล

ออราเคิล [7] คือ โปรแกรมจัดการฐานข้อมูล ผลิตโดยบริษัทออราเคิล ซึ่งเป็นโปรแกรมจัดการฐานข้อมูลเชิงสัมพันธ์ ตัวโปรแกรมนี้จะทำหน้าที่เป็นตัวกลางคอยติดต่อ ประสานระหว่างผู้ใช้และฐานข้อมูล ทำให้ผู้ใช้งานสามารถใช้งานฐานข้อมูลได้สะดวกขึ้น เช่น การค้นหาข้มูลต่างๆ ภายในฐานข้อมูลที่ง่ายและสะดวก โดยผู้ใช้ไม่จำเป็นต้องทราบถึงโครงสร้างภายในของฐานข้อมูลก็สามารถเข้าใช้ฐานข้อมูลนั้นได้

ข้อดีของ ออราเคิล

1) เทคโนโลยี Rollback Segment ถูกนำมาใช้ในโปรแกรมออราเคิล ประโยชน์ Rollback Segment คือ สามารถจัดการกับข้อมูลในกรณีที่เกิดการล้มเหลวของระบบ หรือภาวะระบบไม่สามารถให้บริการได้ ด้วยเทคโนโลยี Rollback Segment จะจัดการ Instance Recovery ข้อมูลไม่ให้เกิดความเสียหายอันเนื่องมาจาก การล้มเหลวของระบบ ได้อย่างดีมาก

2) ออราเคิลยังมีส่วนที่เรียกว่า Timestamp ทำงานเกี่ยวข้องกับ Concurrency Control เป็นส่วนที่จัดการการทำงานกับหลายๆ Transaction ในเวลาเดียวกัน โดยทุกๆ Transaction จะมี Timestamp เป็นตัวกำหนดเวลาเริ่มต้นของการประมวลผล (Process) ซึ่งช่วยในการขจัดปัญหาหลักของ Concurrency Problems

3) ออราเคิลใช้ได้กับฐานข้อมูลกว่า 80 แพลตฟอร์ม ซึ่งครอบคลุมเกือบทุกแพลตฟอร์มที่มีอยู่ในปัจจุบัน เริ่มตั้งแต่เครื่องคอมพิวเตอร์บนเมนเฟรม, มินิคอมพิวเตอร์, พีซี บนระบบปฏิบัติการตั้งแต่ Window 9x, NT, Window CE, UNIX, SOLARIS, LINUX โดยที่ในทุกพอร์ตมีโครงสร้างการเหมือนกันๆหมด คำสั่งที่ใช้ก็เป็นแบบเดียวกัน สามารถทำงานร่วมกันได้ สามารถนำข้อมูลจากพอร์ตหนึ่งไปพอร์ตอื่นได้อย่างไม่มีปัญหา

ประเภทของออราเคิล

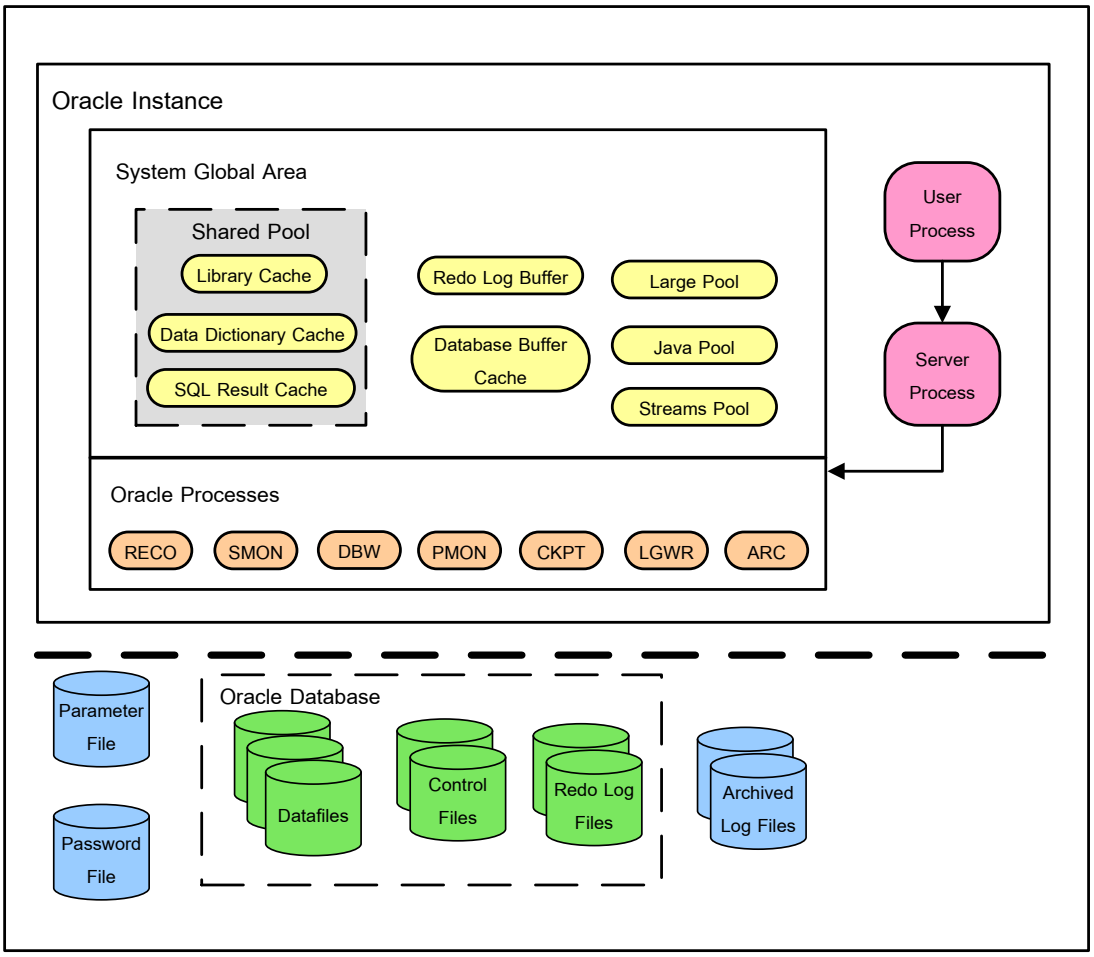
1) ออราเคิลส่วนบุคคล (Personal Oracle)

2) เครื่องแม่ข่ายออราเคิล (Oracle Server)

ทั้ง 2 แบบนี้มีลักษณะการใช้งานและคำสั่งเหมือนกัน แต่ต่างกันที่ออราเคิลส่วนบุคคล คือฐานข้อมูลที่เมื่อติดตั้งบนเครื่องคอมพิวเตอร์แล้ว ผู้ใช้จะต้องนั่งทำงานกับฐานข้อมูลออราเคิลนี้ที่หน้าเครื่องเท่านั้น ส่วนของเครื่องแม่ข่ายออราเคิล คือ ฐานข้อมูลในลักษณะเครื่องแม่ข่ายคือเครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรมของเครื่องแม่ข่ายออราเคิลไว้ และยอมให้ผู้ใช้งานเรียกฐานข้อมูล หรือจัดการกับข้อมูลจากเครื่องคอมพิวเตอร์อื่นๆ ได้ โดยที่เรียกเครื่องอื่นๆ เป็นเครื่องลูกข่าย (Client) ดังนั้นถ้าต้องการให้มีผู้เรียกใช้ข้อมูลจากฐานข้อมูลหลาย ๆ คนได้ ก็ควรต้องเลือกแบบที่เป็นเครื่องแม่ข่ายออราเคิล

สถาปัตยกรรมของออราเคิล

สถาปัตยกรรมระบบจัดการฐานข้อมูลออราเคิล [8] ในการใช้โปรแกรมออราเคิลให้ได้ประสิทธิภาพสูงสุด ผู้ใช้งานต้องเข้าใจถึงโครงสร้างพื้นฐาน สถาปัตยกรรม และการทำงานของฐานข้อมูลออราเคิล ในด้านต่าง ๆ ซึ่งออราเคิลเป็นฐานข้อมูลแรกที่ถูกออกแบบมาให้ทำงานแบบ Grid Computing เพื่อให้เกิดความยืดหยุ่นในการใช้งาน โดยสถาปัตยกรรมฐานข้อมูลของออราเคิลจะประกอบไปด้วยส่วนประกอบหลัก 2 ส่วน คือ Oracle Instance และ Oracle Database ดังภาพที่ 2.4



ภาพที่ 2.4 สถาปัตยกรรมของออราเคิล [8]

จากภาพที่ 2.4 สามารถอธิบายสถาปัตยกรรมของออราเคิลได้ดังนี้

1) Oracle Instance

Oracle Instance คือ หน่วยความจำที่ใช้ในการเก็บข้อมูล และควบคุมการทำงานของ Oracle Sever ที่เรียกว่า SGA(System Global Area) รวมกับ Oracle Process ที่ใช้ในการทำงานของ ออราเคิลรวมเข้าด้วยกัน โดยทุกครั้งเมื่อมีการ Start ฐานข้อมูลขึ้นมา ออราเคิลจะจองพื้นที่หน่วยความจำ (SGA) และสร้าง Oracle Process ขึ้นมาทำงานทุกครั้งเพื่อให้ฐานข้อมูลสามารถทำงานได้ ซึ่งสามารถอธิบายรายละเอียดได้ดังนี้

1.2) SGA (System Global Area)

เป็นหน่วยความจำที่สำคัญที่สุดของออราเคิลทำหน้าที่จัดเก็บข้อมูลและควบคุมการทำงานของ Oracle Sever ให้สามารถทำงานได้ เมื่อเริ่มต้นเปิดการใช้งานฐานข้อมูล (Startup) ออราเคิลจะจองหน่วยความจำนี้ไว้ใช้ในการทำงาน และเมื่อปิดการใช้งาน (Shutdown) หน่วยความจำที่จองไว้นี้จะถูกคืนกลับไปให้กับระบบ ภายใน SGA ประกอบไปด้วย

ส่วนต่าง ๆ ที่ทำงานผสานกัน โดยทำงานเป็นแคช (Cache) ของฐานข้อมูล ซึ่งขึ้นกับหน้าที่ของแต่ละส่วน โดยมีส่วนต่าง ๆ ที่สำคัญมีดังนี้

1.2.1) Shared Pool เป็นหน่วยความจำที่ใช้สำหรับเก็บข้อมูลล่าสุดของฐานข้อมูล เช่น คำสั่ง SQL ที่ใช้งานในฐานข้อมูล Data Dictionary ผลลัพธ์ของคำสั่ง SQL และข้อมูลการทำงานต่าง ๆ (Control Structure) ของฐานข้อมูล สามารถแบ่งหน่วยความจำของ Shared Pool ออกเป็น 3 ส่วนหลัก ๆ ดังนี้

(1) Library Cache เป็นพื้นที่ที่ใช้ในการเก็บคำสั่ง SQL ต่าง ๆ ที่ผู้ใช้งานส่งเข้ามาทำงานในฐานข้อมูล รวมถึง PL/SQL Procedure และ Package ต่าง ๆ ด้วยเช่นกัน นอกจากเก็บคำสั่ง SQL แล้วนั้นในส่วนนี้ยังเป็นพื้นที่ที่ใช้เก็บแผนการทำงานของ SQL Statement ด้วย (Execution Plan) ซึ่งจะเก็บไว้คู่กับ SQL Statement นั้น ๆ จนกว่าจะถูกแทนที่ด้วย Statement อื่น

(2) Data Dictionary Cache เป็นพื้นที่ใช้ในการเก็บข้อมูลของ Data Dictionary ที่ถูกใช้งานล่าสุด เพื่อที่หากต้องการใช้งานข้อมูลนี้อีกจะไม่ต้องไปอ่านมาจากไฟล์ข้อมูลโดยตรง แต่สามารถหาเจอได้ในหน้วยความจำส่วนนี้ได้เลย Data Dictionary จะจัดเก็บข้อมูลของวัตถุ (Object) ต่าง ๆ ว่าประกอบด้วย คอลัมน์อะไรและใครสามารถใช้งานได้บ้าง

(3) SQL Result Cache เป็นหน่วยความจำใหม่ใน Share Pool ที่เพิ่มเข้ามาในเวอร์ชัน 11g โดยใช้เป็นพื้นที่สำหรับเก็บผลลัพธ์ของคำสั่ง SQL และ PL/SQL ที่ส่งเข้ามาในการทำงานฐานข้อมูล ซึ่งเมื่อผู้ใช้งานส่งคำสั่งเดียวกันเข้ามาทำงานในฐานข้อมูลอีกครั้งออราเคิลสามารถหาผลลัพธ์ของ SQL นั้น ๆ จากหน่วยความจำนี้โดยที่ไม่ต้องไปเรียกจากฐานข้อมูล ทำให้การแสดงผลรวดเร็วขึ้น

1.2.2) Database Buffer Cache เป็นหน่วยความจำที่ใช้เก็บข้อมูลล่าสุดที่ถูกใช้งานจากไฟล์ข้อมูลในฐานข้อมูล โดยเมื่อเริ่มต้นการทำงาน ออราเคิลจะตรวจสอบข้อมูลใน Database Buffer Cache ก่อนว่ามีข้อมูลที่ต้องการใช้งานหรือไม่ หากหาไม่เจอออราเคิลจะอ่านข้อมูลขึ้นมาจากไฟล์ข้อมูลและเก็บไว้ใน Database Buffer Cache ดังนั้นเมื่อผู้ใช้รายอื่น ต้องการใช้งานข้อมูลเดียวกันนี้อีกก็สามารถอ่านข้อมูลได้จากหน่วยความจำส่วนนี้ได้ทันที ทำให้ความเร็วในการทำงานมีมากขึ้น นอกจากนี้แล้วการเปลี่ยนแปลงข้อมูลที่ทำใน Database Buffer Cache จะเพิ่มประสิทธิภาพการทำงานให้กับฐานข้อมูลได้ เนื่องจากเป็นการทำงานในส่วนของหน่วยความจำจึงมีความเร็วมากกว่าทำงานกับไฟล์ข้อมูลจริง

1.2.3) Redo Log Buffer เป็นหน่วยความจำที่ใช้เก็บข้อมูลในการทำงานของ Transaction ต่างๆที่เข้ามาทำงานในฐานข้อมูลและมีการเปลี่ยนแปลงข้อมูลในฐานข้อมูล ข้อมูลใน Redo Log Buffer นี้จะถูกบันทึกเก็บไว้ใน Redo Log File โดย Background Process เพื่อเก็บไว้ใช้ในการ Recovery ฐานข้อมูลต่อไป การเปลี่ยนแปลงที่เกิดขึ้นในฐานข้อมูลจะมีหมายเลขกำกับเสมอ เรียกว่า System Change Number (SCN) การใช้งาน Redo Log Buffer จะมีลักษณะเป็นแบบ Circular Buffer คือ ลำดับการเขียนข้อมูลจะเขียนเรียง โดยเรียงตามลำดับก่อนหลังจนกระทั่งเต็มและวนกลับมาเขียนทับข้อมูลเดิม เมื่อมีการเขียนข้อมูลใน Redo Log Buffer จะเริ่มเขียนจากจุดเริ่มต้นไปเรื่อย ๆ จนกระทั่งถึงจุดสุดท้ายแล้วจะกลับมาเริ่มเขยนที่จุดแรกอีกครั้ง โดยระหว่างที่มีการเขียนข้อมูลใน Redo Log Buffer อยู่นั้น จะมี Background Process คอยอ่านข้อมูลจาก Buffer นี้เพื่อบันทึกเก็บไว้ใน Redo Log File อีกทีหนึ่ง การที่ข้อมูลใน Redo Log Buffer จะสามารถถูกวนกลับมาเขียนทับได้อีกครั้งนั้น ข้อมูลนั้นจะต้องถูกบันทึกลงใน Redo Log File เรียบร้อยแล้วจึงจะสามารถถูกเขียนทับได้

1.2.4) Large Pool เป็นหน่วยความจำที่ใช้สำหรับการทำงานที่ต้องใช้พื้นที่ในหน่วยความจำขนาดใหญ่ ได้แก่ การทำงานแบบ Multithread Sever การทำงานของ I/O Process และการทำ Backup ฐานข้อมูลของ Recovery Manager (RMAN) ซึ่งจำเป็นต้องใช้หน่วยความจำจำนวนมากในการทำงาน

1.2.5) Java Pool เป็นหน่วยความจำที่ใช้สำหรับการทำงานของโปรแกรมที่เป็นจาวาต่าง ๆ ที่ผู้ใช้งานส่งเข้ามาทำงานในฐานข้อมูล

1.2.6) Streams Pool เป็นหน่วยความจำที่ใช้เฉพาะสำหรับ Oracle Stream โดยเป็นส่วนที่ใช้เก็บ Query Message ต่าง ๆ ที่ใช้ในการทำงานของ Oracle Streams

1.3) Oracle Process

Oracle Process มีหน้าที่ทำงานในฟังก์ชันต่าง ๆ ของฐานข้อมูล ซึ่ง Process เหล่านี้ประกอบด้วย Background Process ได้แก่

1.3.1) Database Write Process (DBWn) เป็น Process ที่ทำหน้าที่เขียนข้อมูลจาก Database Buffer Cache ลงในไฟล์ข้อมูลที่เป็น Physical File ในฐานข้อมูล ซึ่ง Database Write Process จะเขียนข้อมูลเฉพาะข้อมูลใน Database Buffer Cache ที่มีการเปลี่ยนแปลง (Dirty Buffer) ลงในไฟล์ข้อมูลเท่านั้น ซึ่งจะใช้วิธีการเขียนข้อมูลมากกว่า 1 block เสมอ เพื่อช่วยเพิ่มความเร็วในการทำงาน ขณะที่ฐานข้อมูลเปิดใช้งานอยู่นั้น Database Write Process จะทำงานก็ต่อเมื่อ Server Process ไม่สามารถหาค่าว่างใน Database Buffer Cache

ได้หลังจากที่ทำการ Scan block ไปจนถึงค่าที่กำหนดไว้ และเมื่อถึงช่วงระยะเวลาที่กำหนดเพื่อให้ข้อมูลที่เขียนลงไฟล์นั้นอยู่ก่อนจุด checkpoint เสมอ เพื่อให้เกิดความมั่นใจว่าเมื่อเกิด checkpoint แล้วข้อมูลในไฟล์จะปรับปรุงอยู่เสมอ รวมทั้งลดระยะเวลาในการทำ checkpoint ของฐานข้อมูลให้เร็วขึ้นอีกด้วย

1.3.2) Log Writer Process (LGWR) เป็น Process ที่ทำหน้าที่เขียนข้อมูลใน Log Buffer เก็บลงใน Redo Log File ขณะที่ฐานข้อมูลเปิดใช้งานอยู่นั้น Log Write Process จะทำงานโดยเขียนข้อมูลจาก Log Buffer ลงใน Redo Log File ก็ต่อเมื่อเกิดเหตุการณ์เหล่านี้ขึ้น คือ

(1) เมื่อ Transaction ทำงานเสร็จและสั่ง Commit

(2) ทุกๆ 3 วินาที

(3) พื้นที่ของ Log Buffer ใช้งานไปแล้ว 1 ใน 3 ของทั้งหมด

(4) ก่อนที Database Writer Process จะเขียน Dirty Block ลงในไฟล์ข้อมูล

1.3.3) System Monitor Process (SMON) เป็น Process ที่ทำหน้าที่ตรวจสอบความถูกต้องของข้อมูลในฐานข้อมูล (Data Consistency) ว่าจำเป็นต้องมีการ Recovery หรือไม่เมื่อ Startup ฐานข้อมูลขึ้นมาทุกครั้ง

1.3.4) Process Monitor Process (PMON) เป็น Process ที่ทำหน้าที่ตรวจสอบการทำงานของ Process ต่าง ๆ ในฐานข้อมูล หากมี Process ไหนก็ตามทำงานไม่สำเร็จหรือ Fail ไปในระหว่างที่ทำงานอยู่ PMON จะทำหน้าที่คืนค่าของทรัพยากรต่าง ๆ ที่ Process นั้นจองไว้ใช้งานให้แก่ระบบเพื่อให้สามารถนำทรัพยากรนั้นไปให้กับ Process อื่นที่ต้องการได้

1.3.5) Checkpoint Process (CKPT) เป็น Process ที่ทำหน้าที่ Checkpoint ข้อมูลในฐานข้อมูลให้เกิดความตรงกันของทุก ๆ ไฟล์ในฐานข้อมูล เพื่อเป็นจุดที่จะนำมาใช้อ้างอิงในการ Recovery ฐานข้อมูล โดยเมื่อเกิด Checkpoint ขึ้นในฐานข้อมูล Checkpoint Process จะทำหน้าที่เปลี่ยนแปลงค่าในไฟล์ส่วนหัว (Header File) ของทุกไฟล์ในฐานข้อมูลร่วม Control File ให้ทราบถึงกรกระทำของ Checkpoint ที่เกิดขึ้น เพื่อให้ข้อมูลเกิดความถูกต้องมากที่สุด

1.3.6) Recoverer Process (RECO) เป็น Process ที่ทำหน้าที่จัดการเกี่ยวกับ Distributed Database โดยเมื่อ Transaction มีการทำงานแบบ Distributed

Transaction คือ มีการทำงานกับฐานข้อมูลมากกว่า 1 ฐานข้อมูล Recoverer Process จะทำหน้าที่ติดต่อไปยังฐานข้อมูลอื่น ๆ ที่เกี่ยวข้องเพื่อทำงานใน Transaction นั้นให้สำเร็จ

1.3.7) Archiver Process (ARC) เป็น Process ที่ทำหน้าที่คัดลอก Redo Log File เก็บไว้ใน Archive File เพื่อใช้ในการ Recovery ฐานข้อมูล โดย Archive File จะมีเนื้อหาเหมือนกับข้อมูลใน Redo Log File ทุกประการ การที่ Archive File สามารถช่วยในการ Recovery ฐานข้อมูลได้ เนื่องจาก Archive File เป็นไฟล์ที่คัดลอกมาจาก Redo Log File ซึ่งเก็บรายละเอียดของ Transaction ที่มาเปลี่ยนแปลงข้อมูลในฐานข้อมูลไว้ เมื่อฐานข้อมูลเกิดข้อผิดพลาดและต้องการ Recovery ฐานข้อมูล ออราเคิลจะอ่านค่าการเปลี่ยนแปลงนี้ได้จาก Redo Log File และ หากข้อมูลที่ต้องการไม่อยู่ใน Redo Log File และ จะอ่านค่ามาจาก Archive File ที่เก็บไว้นั่นเอง Archive Fileจะถูกสร้างขึ้นมาก็ต่อเมื่อฐานข้อมูลนั้นมีโหมดการทำงานเป็นแบบ Archive Mode คือ กำหนดให้มีการคัดลอก Redo Log File เก็บไว้ เมื่อ Redo Log File ที่กำลังใช้งานอยู่นั้นถูกเขียนเต็มแล้ว Archive Process จะคัดลอก Redo Log File เก็บไว้ใน Archive File ซึ่งจะมีชื่อแตกต่างกันและจะไม่มีการเขียนไฟล์ทับกัน

นอกจาก Background Process ที่กล่าวมาแล้วนั้น ยังมี Process อีก 2 Process ที่ไม่ได้รวมเป็น Oracle Process แต่ถือว่ามีความสำคัญ คือ

1.3.8) User Process เป็น Process ของผู้ใช้งานที่ทำงานตามโปรแกรมที่กำหนด ซึ่ง Process นี้จะทำงานที่เครื่องลูกข่ายที่เรียกโปรแกรมนั้น ๆ ขึ้นมาทำงาน เมื่อผู้ใช้งานส่งคำสั่งที่ต้องการติดต่อฐานข้อมูล User Process จะส่งผ่านคำสั่งของผู้ใช้งานต่อไปยัง Server Process ที่ฝั่งแม่ข่ายอีกทีหนึ่ง จากนั้นก็จะรอรับผลลัพธ์ของการทำงานที่จะส่งมาจาก Server Process และแสดงให้ผู้ใช้งานทราบ

1.3.9) Server Process เป็น Process ที่ทำงานอยู่ที่ฝั่งเครื่องแม่ข่าย ถูกสร้างขึ้นมาเพื่อรองรับการทำงานของ User Process โดยเมื่อ User Process ที่เครื่องลูกข่ายส่งคำสั่งเข้ามาทำงานในฐานข้อมูล Server Process ที่ฝั่งเครื่องแม่ข่ายจะจับคู่กับ User Process นั้นและจะนำคำสั่งที่ได้ไปทำงานในฐานข้อมูล เมื่อทำงานเสร็จและได้ผลลัพธ์จะส่งผลลัพธ์นั้นกลับคืนให้กับ User Process อีกทีหนึ่ง การทำงานของ Server Process และ User Process จะเป็นการทำงานแบบ 1:1 หรือแบบ Many : 1 แล้วแต่จะกำหนด

2) Oracle Database

Oracle Database คือ กลุ่มข้อมูลที่ถูกจัดเก็บรวมกันเป็นไฟล์ข้อมูลที่เป็น Physical File โดยข้อมูลเหล่านี้สามารถที่จะเรียกขึ้นมาดูหรือเปลี่ยนแปลงค่าได้ ไฟล์ข้อมูลที่ประกอบกันเป็น Oracle Database ประกอบด้วย 3 แบบ คือ

2.1) Control File เป็น Binary File ที่ใช้ควบคุมการทำงานของฐานข้อมูล ถูกสร้างขึ้นมาพร้อมกับการสร้างฐานข้อมูล ขนาดของไฟล์จะเพิ่มขึ้นตลอดเวลา เนื่องจากจะจัดเก็บโครงสร้างและรายละเอียดหลักของฐานข้อมูลทั้งหมดไว้ในไฟล์นี้

2.2) Redo Log File เป็น Binary File ที่เก็บบันทึกรายละเอียดการทำงานของ Transaction ต่าง ๆ ที่ทำงานและเปลี่ยนแปลงข้อมูลในฐานข้อมูล เพื่อใช้ในการ Recovery ฐานข้อมูล

2.3) Data File เป็นไฟล์ข้อมูลที่เก็บข้อมูลต่าง ๆ ของฐานข้อมูล ซึ่ง Data File นี้เป็นส่วนหนึ่งของ Tablespace ซึ่งใช้ในการเก็บข้อมูลของแต่ละผู้ใช้งานในฐานข้อมูล Tablespace นี้จัดเป็น Logical Structure ของฐานข้อมูล

นอกจากนี้ยังมีไฟล์อีก 3 ประเภทที่ไม่ได้เก็บข้อมูลของฐานข้อมูล แต่มีความจำเป็นในการทำงานของฐานข้อมูล คือ

2.4) Parameter File เป็นไฟล์ที่เก็บค่าพารามิเตอร์ต่าง ๆ ที่ใช้ในการทำงานของฐานข้อมูล เมื่อ Startup ฐานข้อมูล Oracle Process จะอ่านข้อมูลจากไฟล์นี้เพื่ออ่านค่าพารามิเตอร์ต่าง ๆ ที่จำเป็นต้องใช้ในการทำงาน

2.5) Archive Log File เป็นไฟล์ที่ได้จากการคัดลอกข้อมูลของ Redo Log File ในฐานข้อมูล โดยจะใช้เป็นไฟล์ที่เก็บสำรองข้อมูลของ Redo Log File เพื่อนำไปใช้ในการ Recovery ฐานข้อมูล เนื่องมาจาก Redo Log File เป็นไฟล์ที่เก็บรายละเอียดการทำงานของ Transaction ในฐานข้อมูล เมื่อฐานข้อมูลเกิดการทำงานผิดพลาดและต้องการ Recovery ฐานข้อมูล ออราเคิลจะอ่านข้อมูลจากการเปลี่ยนแปลงที่เกิดขึ้นในฐานข้อมูลจาก Redo Log File ก่อน และหากหาข้อมูลที่ต้องการไม่เจอ ก็จะมาอ่านไฟล์จาก Archive Log File

2.6) Password File เป็นไฟล์ที่ใช้ในการเก็บรหัสผ่านของผู้ใช้งานที่ต้องการเข้าใช้งานฐานข้อมูล

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 งานวิจัยเรื่อง Optimization of Bulk Operation Performances Within Oracle Database

งานวิจัยนี้ [9] นำเสนอการทดสอบประสิทธิภาพของกระบวนการ Bulk Operation ซึ่งเป็นเทคนิคในการลดภาระของการทำงานแบบวนรอบที่เกิดขึ้นในการส่งผ่านข้อมูลระหว่างส่วนประมวลคำสั่งเอสคิวแอลกับส่วนประมวลคำสั่ง PL/SQL โดยเฉพาะในกรณีที่ข้อมูลมีขนาดใหญ่หรือมีปริมาณมาก ซึ่งจะต้องมีการจำกัดปริมาณข้อมูลที่จะส่งผ่านในแต่ละครั้ง โดยงานวิจัยนี้มีวัตถุประสงค์ที่จะทดสอบประสิทธิภาพของกระบวนการดังกล่าวอันเนื่องมาจากปัจจัยต่างๆ ที่เกี่ยวข้อง ได้แก่

1) ปริมาณของข้อมูลที่ต้องทำการประมวลผล
2) ผลกระทบอันเนื่องมาจากการจัดสรรโครงสร้างหน่วยความจำของออราเคิล ได้แก่ ส่วนของ System Global Area (SGA) และ Process Global Area (PGA)
3) ความแตกต่างเชิงกายภาพของเครื่องคอมพิวเตอร์ที่ใช้ เช่น ความเร็วของหน่วยประมวลผลกลาง (CPU) เป็นต้น
4) ชนิดของข้อมูลที่ต้องทำการประมวลผล

ในการทดสอบแต่ละประเด็นข้างต้น ผู้วิจัยจัดชุดทดสอบที่ปรับเปลี่ยนความแตกต่างในส่วนของแต่ละปัจจัยเหล่านั้น และทำการวัดเวลาที่ใช้ในการประมวลผลเพื่อให้ทราบถึงผลกระทบต่อประสิทธิภาพของ Bulk Operation อันเนื่องมาจากแต่ละปัจจัย ทั้งนี้ การวัดเวลาประมวลผล มีการทดลองซ้ำๆ โดยใช้ชุดทดสอบเดิมหลายครั้ง แล้วจึงหาค่าเฉลี่ยเพื่อลดอิทธิพลของกระบวนการเบื้องหลัง (Background Process) ที่ทำงานอยู่เบื้องหลังในเครื่องคอมพิวเตอร์

ผลที่ได้จากงานวิจัยนี้แสดงให้เห็นว่า ปัจจัยที่ควบคุมประสิทธิภาพของ Bulk Operation ขึ้นอยู่กับขนาดของ SGA เป็นสำคัญ โดยปริมาณของข้อมูลที่ส่งได้สูงสุดนั้น แปรผันไปตามขนาดของ SGA ในขณะที่ขนาดของ PGA, ชนิดของข้อมูล ตลอดจนความแตกต่างเชิงกายภาพของเครื่องคอมพิวเตอร์ เช่น ความเร็วของหน่วยประมวลผลกลางไม่ได้เป็นปัจจัยสำคัญแต่ประการใด

จากงานวิจัยนี้ แม้จะเป็นการวิเคราะห์ประสิทธิภาพของเทคนิคเฉพาะด้านของออราเคิล แต่การกระจายการทดสอบออกตามปัจจัยด้านต่างๆ ที่เกี่ยวข้อง ตลอดจนรูปแบบการทดสอบโดยใช้ชุดทดสอบเดิมซ้ำๆ หลายครั้งเพื่อลดผลกระทบของกระบวนการเบื้องหลังดังกล่าว สามารถเป็น

ตัวอย่างเพื่อนำมาปรับใช้กับการวัดประสิทธิภาพของระบบที่ใช้ฐานข้อมูลออราเคิลที่จะใช้ในวิทยานิพนธ์ฉบับนี้ได้เป็นอย่างดี

2.2.2 งานวิจัยเรื่อง Analysis of the Query Plan for Assessing Optimizer Performance of SQL Select Statement

งานวิจัยนี้ [10] นำเสนอการวิเคราะห์ และประเมินถึงประสิทธิภาพการทำงานของดีบีเอ็มเอสที่มีต่อคำสั่งเอสคิวแอล โดยใช้แผนการสืบค้น (Query Plan) ซึ่งจะแบ่งการทำงานออกเป็น 4 ขั้นตอนคือ 1) จัดเตรียมเครื่องมือและข้อมูลการทดสอบ โดยใช้ออราเคิล เวอร์ชัน 11.1 และ SQL Developer เวอร์ชัน 1.5.5 2) ออกแบบคำสั่งเอสคิวแอล 3) ประมวลผลคำสั่งเอสคิวแอลและสร้างแผนการสืบค้น และ 4) วิเคราะห์และสรุปผลการทำงาน ผลของงานวิจัยนี้สรุปได้ว่า การสืบค้นข้อมูลในตารางขนาดใหญ่ผู้ใช้ควรกำหนดให้อ๊อพติไมเซอร์ลิงค์ข้อมูลด้วยการกำหนดอ๊อพชัน Hash Join ผ่านคำสั่ง /*Hint “use_has” */ จะช่วยให้ประมวลผลได้เร็วขึ้น และการเพิ่มประสิทธิภาพการประมวลผลคำสั่งเอสคิวแอลนั้นขึ้นอยู่กับโครงสร้างและรูปแบบคำสั่งด้วย

จากงานวิจัยนี้สามารถนำมาใช้เป็นตัวอย่างในการปรับปรุงประสิทธิภาพในวิทยานิพนธ์นี้ได้เป็นอย่างดี

2.2.3 งานวิจัยเรื่อง Non-linear Optimization of Performance Functions for Autonomic Database Performance Tuning

งานวิจัยนี้ [11] ได้นำเสนอรูปแบบการอ้างอิงตามองค์ประกอบความรู้ที่มีความสำคัญในการสร้างของเครื่องมือการจัดการอัตโนมัติความสามารถในการตระหนักถึงการ MAPE การควบคุมความคิดเห็น พัฒนารูปแบบการอ้างอิงทั่วไปที่จับขึ้นต่อกันระหว่างเซ็นเซอร์ของระบบจัดการฐานข้อมูล (DBMS : Database Management System) กับตัวผลกระทบ และสามารถนำมาใช้ในกระบวนการของการปรับแต่งประสิทธิภาพของฐานข้อมูลอัตโนมัติ นอกจากนี้ยังให้ภาพรวมมากกว่าการพัฒนาต้นแบบ เพื่อที่จะรวบรวมข้อมูลระบบเฉพาะสำหรับรูปแบบการอ้างอิงและประเมินผลโดยการกำหนดและการแก้ปัญหาการปรับปรุงประสิทธิภาพ

2.2.4 งานวิจัยเรื่อง Self-Tuning for SQL Performance in Oracle Database 11g

งานวิจัยนี้ [12] นำเสนอถึงออราเคิลเวอร์ชัน 11 จี ที่มีการพัฒนาศักยภาพที่สูงขึ้น โดยมีการทำงานแบบอัตโนมัติและมีระบบป้องกันในตัวเองเพื่อป้องกันผลกระทบในการทำงานแต่ละครั้ง ซึ่งผู้ดูแลฐานข้อมูล (DBA : Database Administrator) สามารถวิเคราะห์ออกมาด้วยตัวเอง

ถึงคำสั่งของเอสคิวแอล และดูผลรวมของศักยภาพที่ได้ปรับปรุง ผลของขั้นตอนการทำงานทุกอย่างจะรายงานออกมาเพื่อดูปัญหาที่เกิดขึ้นกับการทำงานของเอสคิวแอลในแต่ละขั้นตอน เพื่อหาวิธีแก้ไข โดยการปรับแต่งเอสคิวแอล คือ การวิเคราะห์สถิติโดยรวมรวมสถิติจากข้อมูลที่อัพเดทของ CBO การรวบรวมข้อมูลที่เป็นประโยชน์และรายละเอียดของเอสคิวแอล วิเคราะห์จากค่าดรรชนี โดยดูจากดรรชนีที่หายไป ดูข้อมูลที่ดรรชนีชี้วัดในขั้นตอนที่จะทำให้กระบวนการทำงานบรรลุผลได้ ใช้กลยุทธ์ด้านเวลาในการทำงานของการปรับแต่งเอสคิวแอลโดยการกำหนดจุดการทำงาน หาข้อบกพร่อง แก้ไข จนกระทั่งประสบความสำเร็จ และรายงานผลที่เกิดขึ้น ซึ่งวิธีการดำเนินงานของออราเคิลเวอร์ชัน 11 จี คือ 1) ขอบข่ายงานสำหรับการซ่อมบำรุงแบบอัตโนมัติ หรือที่เรียกว่า Autotask ซึ่งทำให้ระบบการทำงานแบบเป็นหน่วยเดียวกัน (Autotask และ SQL) 2) ตั้งเป้าหมายของการทำงานและการหาค่าที่เหมาะสมที่สุดของจุดมุ่งหมาย โดยอาจขึ้นอยู่กับ การสนับสนุนในการตัดสินใจปรับเปลี่ยน (DSS) และขั้นตอนในการดำเนินการติดต่อออนไลน์ (OLTP) 3) การคัดเลือกคำสั่งเอสคิวแอล โดยดูจากระยะเวลาและประสิทธิภาพในการทำงานเป็นหลัก 4) การทดลองระบบเอสคิวแอล โดยสังเกตจากการใช้งานจริง เพื่อสำรวจหาข้อบกพร่อง และหาแนวทางแก้ไข 5) กฎที่ต้องมีขึ้นเพื่อการปรับปรุง จากการวิเคราะห์อย่างน้อยจะต้องมีการปรับปรุงเดือนละครั้ง 6) รายงานผลสำหรับการจัดการและดำเนินการทั้งหมด

2.2.5 **งานวิจัยเรื่อง** Knowledge Management in Autonomic Database Performance Tuning

งานวิจัยนี้ [13] เสนอถึงความรู้ในส่วนประกอบของฐานข้อมูลที่เป็นที่ต้องการสำหรับการบริหารจัดการในโครงสร้างแบบอัตโนมัติและมีความเข้าใจได้ดีในตัวของ MAPE นอกจากนั้นเรายังพัฒนาการพึ่งพาอาศัยกันในระบบเอฟเฟคเตอร์ (Effectors) และเซ็นเซอร์ (Sensors) สามารถใช้วิธการแบบอัตโนมัติในรูปแบบของการปรับแต่งประสิทธิภาพอีกด้วย การวิจัยด้านการปรับแต่งประสิทธิภาพของฐานข้อมูลแบบอัตโนมัติเป็นสิ่งที่น่าสนใจแล้ว การนำเครื่องจักรมาใช้แทนคนในการทำงาน (ระบบอัตโนมัติ) เป็นข้อเท็จจริงที่ควรศึกษาและใส่ใจเป็นอย่างยิ่ง

2.2.6 Oracle Enterprise Manager

ออราเคิลเอ็นเตอร์ไพร์สเมเนเจอร์ (Oracle Enterprise Manager) [14] เป็นเครื่องมือที่ใช้ในการบริหารฐานข้อมูลออราเคิล โดยจะแสดงสภาวะการทำงานของฐานข้อมูลออราเคิลช่วงเวลาหนึ่งจนถึงปัจจุบัน, มีกราฟแสดงจำนวนเซสชัน (Session) ที่กำลังใช้หน่วยประมวลผล, จำนวนเซสชันที่กำลังใช้งาน, จำนวนรับเข้าและส่งออก (I/O) และแสดงรายละเอียดของคำสั่งเอสคิวแอลที่มีเวลารอคอยสูงสุด

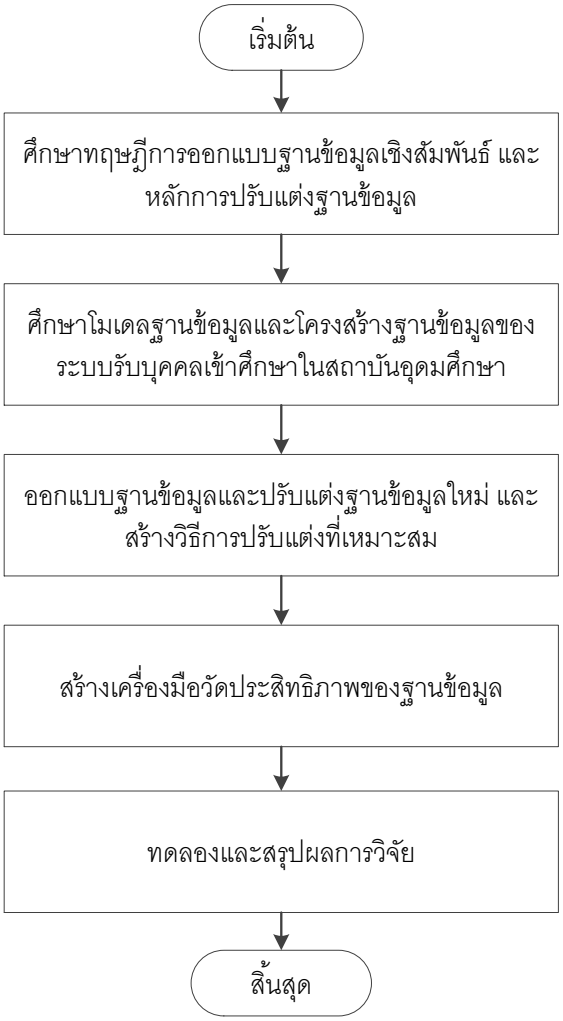
บทที่ 3

แนวคิดและขั้นตอนการวิจัย

แนวคิดและขั้นตอนการวิจัยนั้น เริ่มจากขั้นตอนการดำเนินการวิจัย จากนั้นอธิบายโครงสร้างฐานข้อมูลของระบบ การออกแบบและปรับปรุงประสิทธิภาพของฐานข้อมูล การพัฒนาเครื่องมือเพื่อวัดประสิทธิภาพ การทดสอบการทำงานของโปรแกรมประยุกต์ และการประเมินผลการปรับปรุงประสิทธิภาพ ซึ่งมีรายละเอียดดังต่อไปนี้

3.1 ขั้นตอนการดำเนินการวิจัย

แนวคิดในการสร้างวิธีการปรับแต่งฐานข้อมูลของระบบ คือ สมรรถนะในการทำงานของฐานข้อมูลสามารถดำเนินการได้โดยการปรับแต่ง การปรับแต่งที่มีประสิทธิภาพและเพิ่มสมรรถนะของระบบขึ้นกับลักษณะงานและวิธีการออกแบบของฐานข้อมูล และเนื่องจากมีผู้ใช้ฐานข้อมูลออราเคิลมากขึ้นทุกวัน การที่ได้เรียนรู้และศึกษาวิธีการปรับแต่งประสิทธิภาพจะช่วยให้เข้าใจแนวทางใหม่ๆ เพื่อนำมาปรับปรุงงานให้มีประสิทธิภาพมากยิ่งขึ้น งานวิจัยนี้ได้ทำการปรับปรุงฐานข้อมูลออราเคิลของกรณีศึกษาระบบการรับสมัครบุคคลเข้าศึกษาในสถาบันอุดมศึกษา สามารถวิเคราะห์และออกแบบฐานข้อมูลตามแนวทางการออกแบบเพื่อปรับแต่งฐานข้อมูล ซึ่งการดำเนินงานวิจัยนี้จะแบ่งขั้นตอนดังแสดงในภาพที่ 3.1 และสามารถอธิบายภาพรวมของวิธีการวิจัยดังแสดงในภาพที่ 3.2 ดังต่อไปนี้



ภาพที่ 3.1 แสดงภาพขั้นตอนการดำเนินงานวิจัย

จากภาพที่ 3.1 แสดงขั้นตอนการดำเนินงานวิจัย ดังนี้

3.1.1 ศึกษาทฤษฎีการออกแบบฐานข้อมูลเชิงสัมพันธ์ และหลักการปรับแต่งฐานข้อมูลเพื่อเพิ่มประสิทธิภาพของระบบ

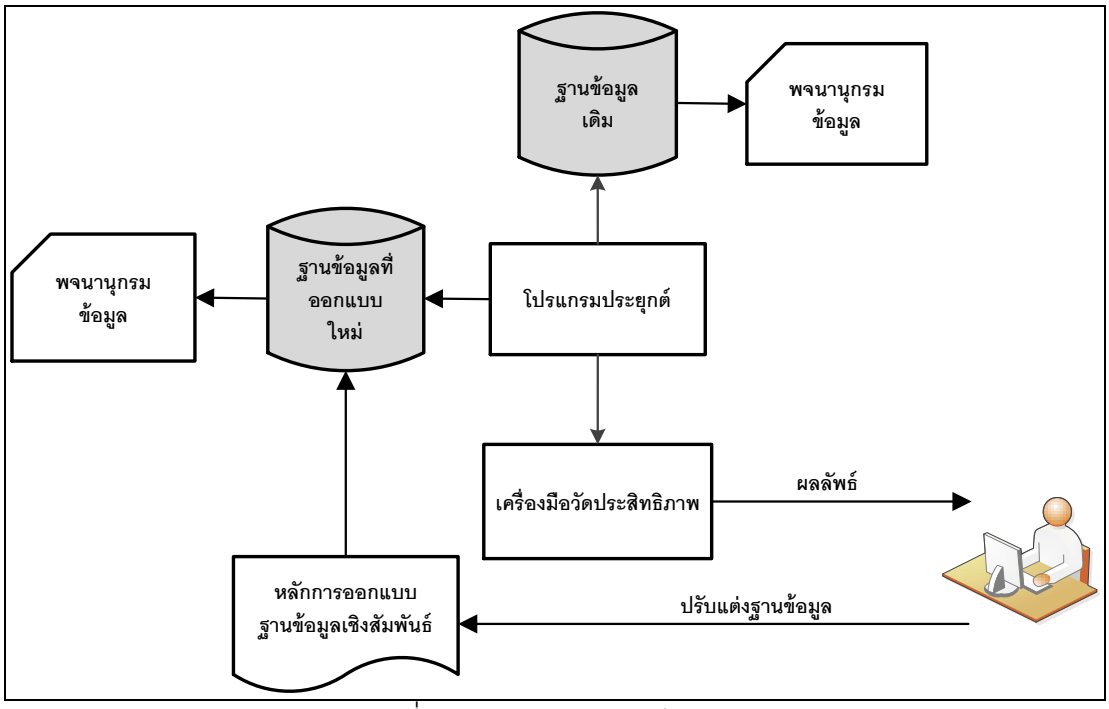
3.1.2 ศึกษาโมเดลฐานข้อมูล และโครงสร้างฐานข้อมูลของกรณีศึกษาระบบรับบุคคลเข้าศึกษาในสถาบันอุดมศึกษาที่ใช้อยู่ในปัจจุบัน จะศึกษาระบบที่ใช้ทดสอบในงานวิจัยนี้ 3 ประกอบย่อย คือ ระบบรับสมัครบุคคลเข้าศึกษาในสถาบันอุดมศึกษา ระบบรวมคะแนนและจัดลำดับการคัดเลือก และระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

3.1.3 ออกแบบฐานข้อมูลและปรับแต่งฐานข้อมูลใหม่ พร้อมทั้งสร้างวิธีการปรับแต่งที่เหมาะสมสำหรับระบบงานในลักษณะของกรณีศึกษานี้

3.1.4 สร้างเครื่องมือวัดประสิทธิภาพของฐานข้อมูล ทั้งนี้สำหรับงานวิจัยนี้ใช้เครื่องมือในการพัฒนา ได้แก่ ฐานข้อมูลออราเคิล ภาษาพีเอชพี (PHP) และใช้ซอฟต์แวร์แมโครมีเดีย ดรีมวีฟเวอร์ (Macromedia Dreamweaver) เวอร์ชัน 8 เพื่อเป็นการแสดงผลประสิทธิภาพของฐานข้อมูลเฉพาะส่วนได้ตรงตามเป้าหมาย

3.1.5 ทดลองและสรุปผลการวิจัย จะนำผลลัพธ์ที่ได้จากเครื่องมือวัดประสิทธิภาพที่สร้างขึ้นมาเปรียบเทียบระหว่างการออกแบบฐานข้อมูลเดิมกับการออกแบบฐานข้อมูลและปรับแต่งฐานข้อมูลใหม่

สำหรับภาพรวมวิธีการวิจัยสามารถแสดงได้ดังภาพที่ 3.2 ดังนี้



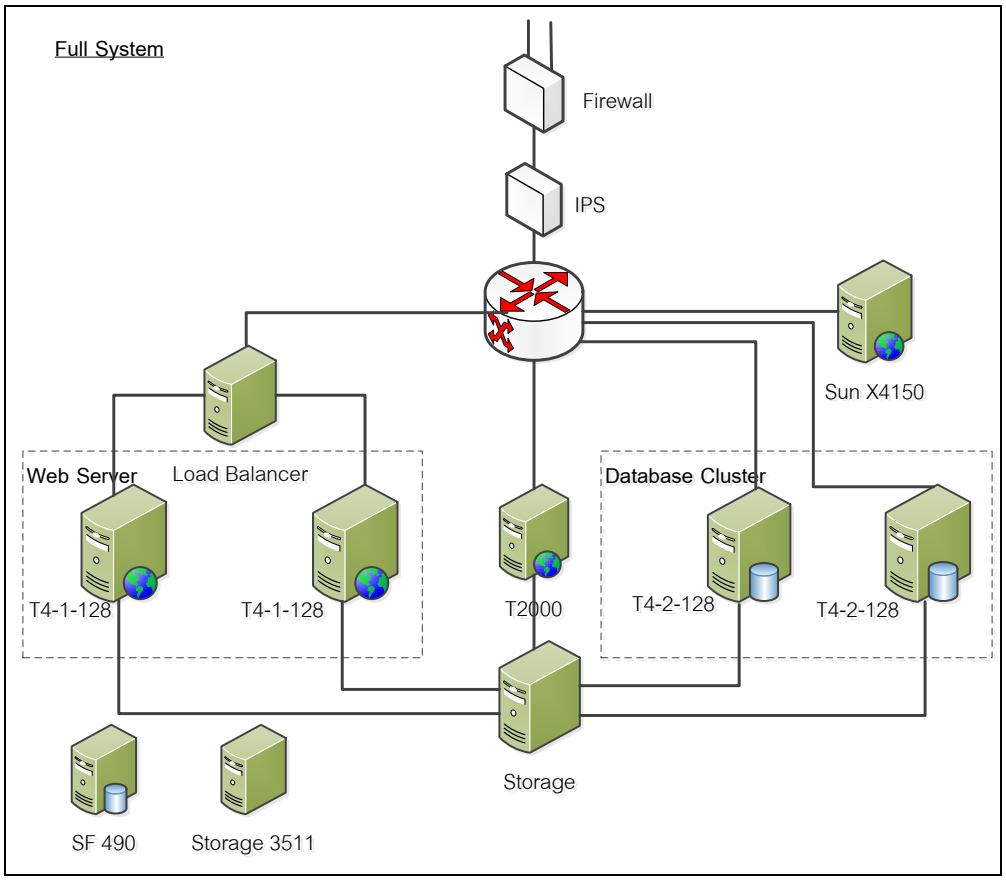
ภาพที่ 3.2 แสดงภาพรวมวิธีการวิจัย

จากภาพที่ 3.2 เป็นภาพรวมของวิธีการวิจัยที่จะได้มาซึ่งวิธีการในการปรับแต่งฐานข้อมูลของระบบ โดยเริ่มจากผู้ใช้ทำการปรับแต่งฐานข้อมูลและสร้างวิธีการปรับแต่งที่เหมาะสมตามหลักการออกแบบฐานข้อมูลเชิงสัมพันธ์เพื่อเพิ่มประสิทธิภาพให้กับการทำงานของโปรแกรมประยุกต์ที่เชื่อมต่อกับฐานข้อมูลออราเคิล จากนั้นวัดประสิทธิภาพของฐานข้อมูลเดิมและฐานข้อมูลที่ออกแบบใหม่ด้วยเครื่องมือวัดประสิทธิภาพ จากนั้นทดลองปรับแต่งฐานข้อมูลไปเรื่อยๆ จนกระทั่งได้ผลลัพธ์ตามที่ต้องการ แล้วนำผลลัพธ์ที่ได้ไปประเมินประสิทธิภาพของฐานข้อมูลที่ได้ปรับปรุงขึ้น ซึ่งจะอธิบายรายละเอียดในลำดับต่อไป

3.2 โครงสร้างฐานข้อมูลของระบบ

ในส่วนนี้ได้ทำการศึกษาโมเดลฐานข้อมูลและโครงสร้างฐานข้อมูลของระบบรับบุคคลเข้าศึกษาในสถาบันอุดมศึกษาที่ใช้อยู่ในปัจจุบัน พร้อมทั้งศึกษาและวิเคราะห์ระบบที่จะใช้ทดสอบในงานวิจัยนี้ 3 ประกอบย่อย คือ ระบบรับสมัคร ระบบรวมคะแนน และระบบประกาศผลการคัดเลือกฯ ซึ่งได้ดำเนินการศึกษาและวิเคราะห์ ดังนี้

3.2.1 ศึกษาและวิเคราะห์โครงสร้างของระบบที่ใช้งานในปัจจุบัน มีโครงสร้าง ดังภาพที่ 3.3 ดังนี้



ภาพที่ 3.3 แสดงโครงสร้างของระบบที่ใช้งานในปัจจุบัน

จากภาพที่ 3.3 โครงสร้างของระบบประกอบด้วย Firewall, IPS, Load Balancer, Web Server (T4-1-128 จำนวน 2 เครื่อง), Storage, SF490, Sun X4150, T2000 และ Database Cluster (T4-2-128 จำนวน 2 เครื่อง) ลักษณะการทำงานของระบบจะมีแอปพลิเคชันอยู่ที่เครื่องแม่ข่าย T4-1-128 ทั้ง 2 เครื่อง และฐานข้อมูลอยู่ที่เครื่องแม่ข่าย (Database Cluster) T4-2-128 ซึ่งทั้ง 2 เครื่อง ใช้ฐานข้อมูลออราเคิล เก็บข้อมูลไว้ที่ Storage และการทำงานของเครื่องแม่ข่าย T4-1-128 ผ่าน Load Balancer

จากสภาพแวดล้อมที่ใช้งานในปัจจุบันได้ถูกปรับปรุงโดยเพิ่มเติมฮาร์ดแวร์ใหม่ ซึ่งเป็นวิธีการเพิ่มประสิทธิภาพให้กับระบบด้านฮาร์ดแวร์ แต่การทำงานของโปรแกรมประยุกต์ยังมีความล่าช้าเช่นเดิม

3.2.2 ศึกษาและวิเคราะห์ปัญหาของระบบฐานข้อมูลที่ใช้ในปัจจุบันของระบบการรับบุคคลเข้าศึกษาในสถาบันอุดมศึกษา ปัญหาที่พบได้แก่

3.2.2.1 การใช้คำสั่งเอสคิวแอลที่ผู้พัฒนาเขียนขึ้นและใช้อยู่ในโปรแกรมที่พัฒนานั้นผลลัพธ์ที่ได้ถูกต้อง แต่คำสั่งเอสคิวแอลบางส่วนมีการเขียนที่ผิดรูปแบบแม้จะทำงานได้แต่ใช้เวลาในการประมวลผลนานกว่าที่ควรจะเป็น

3.2.2.2 สร้างดรรชนีไม่ตรงตามคำสั่งเอสคิวแอลที่เขียนขึ้น บางครั้งมีการลบและสร้างตารางใหม่ แต่ไม่ทำการสร้างดรรชนีใหม่ทำให้ดรรชนีที่มีอยู่แล้วไม่ถูกใช้งาน

3.2.2.3 การใช้คำสั่งเอสคิวแอลไม่เป็นไปตามหลักการใช้คำสั่งที่มีประสิทธิภาพสูงสุด

3.2.2.4 ปัญหาสิ่งสิ้นเปลืองที่ไม่จำเป็นของการบันทึกรายการ และปัญหาการกำจัด ข้อมูลที่ไม่จำเป็น

3.2.2.5 ปัญหาขนาดบัฟเฟอร์และตารางเวลาของการประมวลผล เมื่อจำนวนผู้ใช้เข้ามาจำนวนมากในช่วงเวลาประกาศผลการคัดเลือกฯ ของระบบทำให้ขนาดบัฟเฟอร์มากเป็นผลให้การประมวลผลล้มเหลว

3.2.2.6 ตารางข้อมูลมีความซ้ำซ้อนกันมาก บางคอลัมน์ไม่ได้นำมาใช้งานทำให้เสียพื้นที่เก็บข้อมูลเกินความจำเป็น

จากการวิเคราะห์ปัญหาความล่าช้าของฐานข้อมูล การออกแบบและปรับปรุงฐานข้อมูลใหม่เพื่อเพิ่มประสิทธิภาพของระบบ (ตามขั้นตอนที่ 3.3) ซึ่งจากการวิเคราะห์พบว่าปัจจัยที่มีผลต่อปัญหาข้างต้นนั้น ได้แก่

1) ปริมาณของข้อมูลที่ทำการประมวลผล หมายถึง ปริมาณข้อมูลจำนวนมากตั้งแต่อดีตถึงปัจจุบัน (พ.ศ.2541-2555)

2) การจัดสรรหน่วยความจำของฐานข้อมูลออราเคิล หมายถึง ส่วนของหน่วยความจำของฐานข้อมูลออราเคิลที่ใช้อยู่ในปัจจุบันไม่มีการปรับปรุงอย่างสม่ำเสมอ

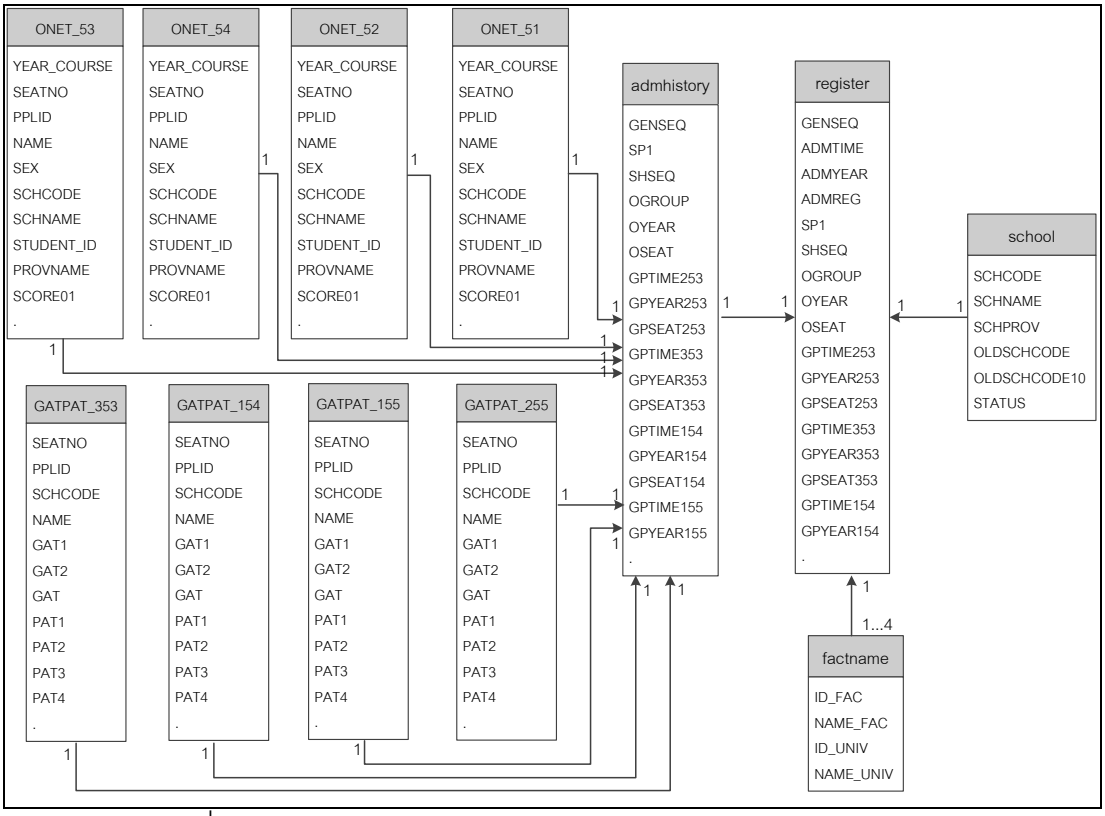
3) ชนิดของข้อมูลที่ทำการประมวลผล กล่าวคือระบบเดิมชนิดของข้อมูลเป็นอักขระ (Character) ทั้งหมด และไม่มีการกำหนดชนิดข้อมูลตามแบบที่ควรจะเป็น เช่น เขตข้อมูลวันเกิดเก็บเป็นอักขระแทนที่จะเป็นชนิดข้อมูลวันที่ (Date) เขตข้อมูลเลขที่สมัครเก็บเป็นอักขระแทนที่จะเป็นชนิดข้อมูลตัวเลข (Number) เป็นต้น

4) การออกแบบฐานข้อมูลที่อ้างอิงจากระบบเดิม หมายถึง ระบบเดิมถูกออกแบบและเก็บข้อมูลแบบแฟ้มข้อมูล ซึ่งส่งผลให้ผู้พัฒนาระบบออกแบบฐานข้อมูลตามรูปแบบของแฟ้มข้อมูลเดิม เช่น ออกแบบให้มีเขตข้อมูลว่างตามรูปแบบแฟ้มข้อมูลเดิม เป็นต้น

5) แอปพลิเคชันแสดงผลลัพธ์ได้ถูกต้อง หมายถึง เมื่อผู้พัฒนาเห็นว่าแอปพลิเคชันที่พัฒนาขึ้นได้ผลลัพธ์ที่ถูกต้องจะยังคงการออกแบบฐานข้อมูลแบบเดิม และใช้คำสั่งเอสคิวแอลตามที่ได้ผลลัพธ์ถูกต้องโดยไม่มีการปรับปรุงประสิทธิภาพของแอปพลิเคชันนั้นๆ เลย

3.2.3 วิเคราะห์โครงสร้างของฐานข้อมูลที่ใช้ในปัจจุบัน จะแบ่งการวิเคราะห์แยกออกเป็น 3 ระบบ ดังนี้

3.2.3.1 ระบบรับสมัครบุคคลเข้าศึกษาในสถาบันอุดมศึกษา มีโครงสร้างของฐานข้อมูลดังภาพที่ 3.4 ดังนี้



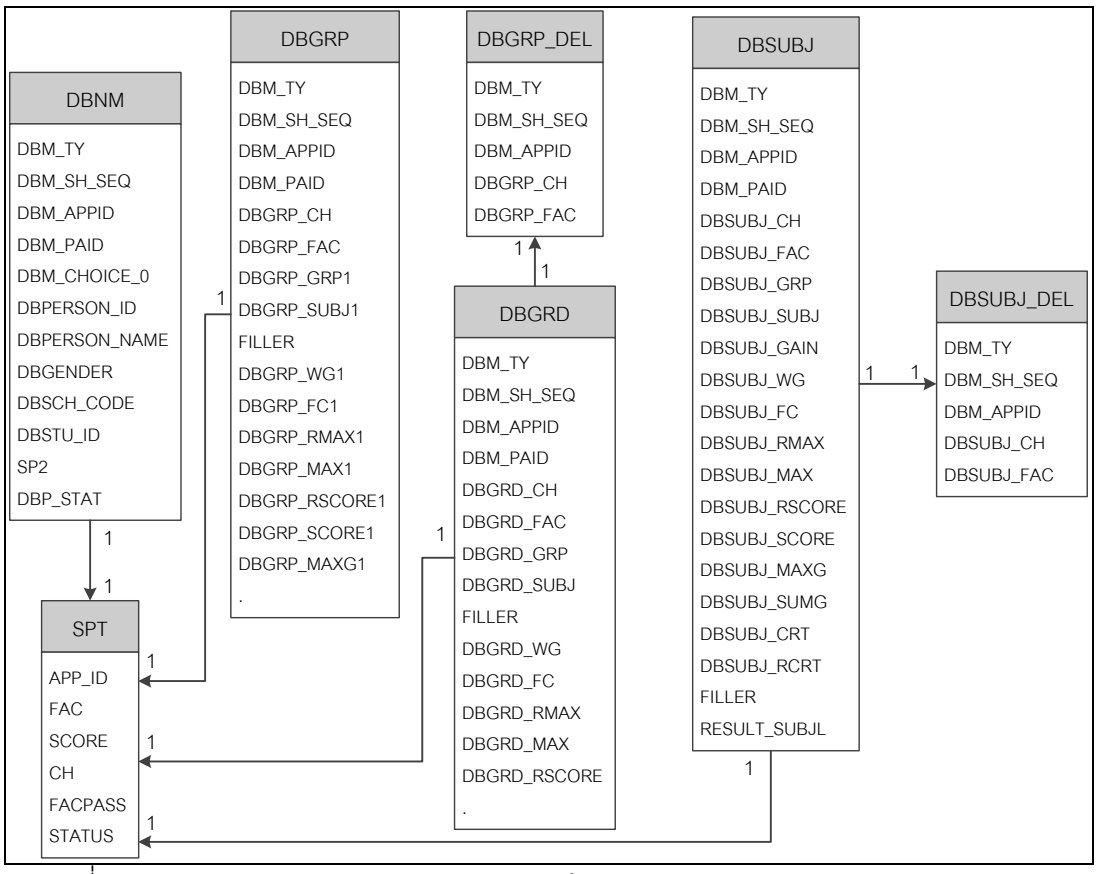
ภาพที่ 3.4 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลของระบบรับสมัคร

จากภาพที่ 3.4 ประกอบด้วยตารางข้อมูล ดังนี้

1. ตาราง admhistory เป็นตารางเก็บข้อมูลประวัติการสอบ O-NET และ GAT/PAT
2. ตาราง register เป็นตารางเก็บข้อมูลผู้สมัคร
3. ตาราง school เป็นตารางเก็บข้อมูลโรงเรียน
4. ตาราง factname เป็นตารงเก็บข้อมูลคณะ/สาขา และมหาวิทยาลัย
5. ตาราง ONET_51 เป็นตารางเก็บข้อมูลคะแนนโอเน็ต ปี 2551
6. ตาราง ONET_52 เป็นตารางเก็บข้อมูลคะแนนโอเน็ต ปี 2552
7. ตาราง ONET_53 เป็นตารางเก็บข้อมูลคะแนนโอเน็ต ปี 2553
8. ตาราง ONET_54 เป็นตารางเก็บข้อมูลคะแนนโอเน็ต ปี 2554
9. ตาราง GATPAT_353 เป็นตารางเก็บข้อมูลคะแนน GATPAT ครั้งที่ 1
10. ตาราง GATPAT_154 เป็นตารางเก็บข้อมูลคะแนน GATPAT ครั้งที่ 2
11. ตาราง GATPAT_155 เป็นตารางเก็บข้อมูลคะแนน GATPAT ครั้งที่ 3
12. ตาราง GATPAT_255 เป็นตารางเก็บข้อมูลคะแนน GATPAT ครั้งที่ 4

สำหรับพจนานุกรมข้อมูล ของแต่ละตารางสามารถดูได้ที่ ภาคผนวก ก

3.2.3.2 ระบบรวมคะแนนและจัดลำดับการคัดเลือก มีโครงสร้างของฐานข้อมูล ดังภาพที่ 3.5 ดังนี้



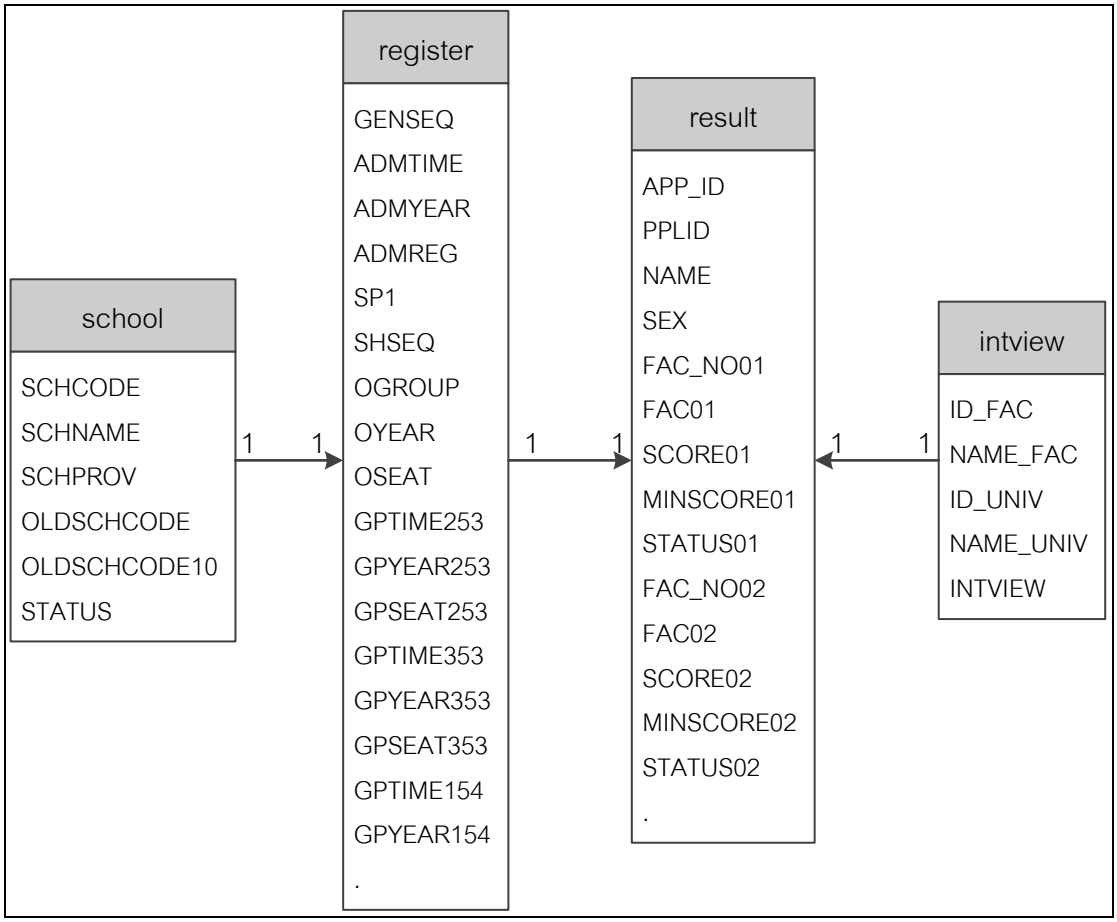
ภาพที่ 3.5 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลของระบบรวมคะแนนและจัดลำดับการคัดเลือก

จากภาพที่ 3.5 ประกอบด้วยตารางข้อมูล ดังนี้

1. ตาราง DBNM เป็นตารางเก็บข้อมูลผู้สมัคร
2. ตาราง DBSUBJ เป็นตารางเก็บข้อมูลคะแนนรายวิชา
3. ตาราง DBGRP เป็นตารางเก็บข้อมูลตามเกณฑ์กลุ่มวิชา
4. ตาราง DBGRD เป็นตารางเก็บข้อมูลคะแนนตามกลุ่มวิชา
5. ตาราง DBSUBJ_DEL เป็นตารางเก็บข้อมูลที่โดนตัดสิทธิ์รายวิชา
6. ตาราง DBGRP_DEL เป็นตารางเก็บข้อมูลที่โดนตัดสิทธิ์ตกเกณฑ์กลุ่มวิชา
7. ตาราง SPT เป็นตารางเก็บข้อมูลผลลัพธ์ที่ใช้ประมวลผล

สำหรับพจนานุกรมข้อมูล ของแต่ละตารางสามารถดูได้ที่ ภาคผนวก ก

3.2.3.3 ระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา มีโครงสร้างของฐานข้อมูลดังภาพที่ 3.6 ดังนี้



ภาพที่ 3.6 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลของระบบประกาศผล

จากภาพที่ 3.6 ประกอบด้วยตารางข้อมูล ดังนี้

1. ตาราง result เป็นตารางเก็บข้อมูลผู้สมัคร คะแนนรวม และผลการคัดเลือกฯ
2. ตาราง school เป็นตารางเก็บข้อมูลโรงเรียน
3. ตาราง intview เป็นตารางเก็บข้อมูลการสอบสัมภาษณ์
4. ตาราง school เป็นตารางเก็บข้อมูลโรงเรียน

สำหรับพจนานุกรมข้อมูล ของแต่ละตารางสามารถดูได้ที่ ภาคผนวก ก

3.3 การออกแบบและปรับปรุงประสิทธิภาพของฐานข้อมูล

สำหรับส่วนนี้จะทำการออกแบบฐานข้อมูลและปรับแต่งฐานข้อมูลใหม่ พร้อมทั้งสร้างวิธีการปรับแต่งที่เหมาะสมของกรณีศึกษานี้ โดยใช้แนวทางการออกแบบ และวิธีการปรับปรุงประสิทธิภาพ ดังนี้

แนวทางที่ 1 สร้างดรรชนีสำหรับเขตข้อมูลที่เรียงลำดับและเงื่อนไข [5,8,15] จากปัญหาที่พบในระบบ คือ การสร้างดรรชนีไม่ตรงตามคำสั่งเอสคิวแอลที่เขียนขึ้น บางครั้งมีการลบและสร้างตารางใหม่แต่ไม่ทำการสร้างดรรชนีใหม่เป็นเหตุทำให้ดรรชนีที่มีอยู่แล้วไม่ถูกใช้งาน ดังนั้นแนวทางนี้จะพิจารณาการสร้างดรรชนีสำหรับเขตข้อมูลเพื่อให้สนับสนุนต่อคำสั่งเอสคิวแอลและเงื่อนไขของคำสั่งนั้นๆ ซึ่งการสร้างดรรชนีควรสร้างตามเงื่อนไขที่ใช้งานจริง เพราะฐานข้อมูลต้องอัพเดตดรรชนีทุกครั้งเมื่อมีผู้ใช้คำสั่งเอสคิวแอล ดังนั้น เมื่อมีการสร้างดรรชนีในตารางหนึ่งไว้มากและซ้ำซ้อนจะทำให้ประสิทธิภาพการทำงานช้าลง จากการวิเคราะห์การสร้างดรรชนีของระบบการรับบุคคลเข้าศึกษาในสถาบันอุดมศึกษาปัจจุบันพบว่ามีการสร้างดรรชนีจำนวนมาก แต่บางดรรชนีไม่มีการใช้งานจริงและมีการสร้างที่ซ้ำซ้อน รายการดรรชนีในระบบดังแสดงในภาพที่ 3.7 ดังนี้

Edit View Delete Actions Create Like Go — Previous 25 51-75 of 240 Next 25

Select	Table Owner	Table	Indexed Columns	Index Owner	Index	Table Type	Tablespace	Partitioned	Last Analyzed
◉	ADM_WEB	ANET_FIELD	CENTERCODE, FIELDCODE	ADM_WEB	ANET_FIELD_IDX3	TABLE	ADM_WEB	NO	Dec 19, 2006 10:00:05 PM GMT+07:00
○	ADM_WEB	ANET_GPAPR	SCHOOLID	ADM_WEB	ANET_GPAPR_IDX1	TABLE	ADM_WEB	NO	Dec 24, 2008 10:09:58 PM GMT+07:00
○	ADM_WEB	ANET_GPAPR	SYEAR, PPLID	ADM_WEB	ANET_GPAPR_IDX2	TABLE	ADM_WEB	NO	Dec 24, 2008 10:10:01 PM GMT+07:00
○	ADM_WEB	ANET_GPAPR	NAME	ADM_WEB	ANET_GPAPR_IDX3	TABLE	ADM_WEB	NO	Dec 24, 2008 10:10:01 PM GMT+07:00
○	ADM_WEB	ANET_GPAPR	SCHOOLID, SYEAR	ADM_WEB	ANET_GPAPR_IDX4	TABLE	ADM_WEB	NO	Dec 24, 2008 10:10:07 PM GMT+07:00
○	ADM_WEB	ANET_GPAPR	PPLID	ADM_WEB	ANET_GPAPR_IDX5	TABLE	ADM_WEB	NO	Dec 24, 2008 10:10:09 PM GMT+07:00
○	ADM_WEB	ANET_SUBJ_ADM	PPLID	ADM_WEB	ANET_PPLID_INDX	TABLE	ADM_WEB	NO	Apr 2, 2008 10:01:10 PM GMT+07:00
○	ADM_WEB	ANET_SUBJ_ADM	REGNO	ADM_WEB	ANET_REGNO_INDX	TABLE	ADM_WEB	NO	Apr 2, 2008 10:01:08 PM GMT+07:00
○	ADM_WEB	ANET_SCHOOL	SCHCODE	ADM_WEB	ANET_SCHOOL_IDX1	TABLE	ADM_WEB	NO	Feb 18, 2008 10:00:09 PM GMT+07:00
○	ADM_WEB	ANET_SCHOOL	OLDSCHCODE	ADM_WEB	ANET_SCHOOL_IDX2	TABLE	ADM_WEB	NO	Feb 18, 2008 10:00:09 PM GMT+07:00
○	ADM_WEB	ANET_SEAT	SEATNO	ADM_WEB	ANET_SEAT_IDX1	TABLE	ADM_WEB	NO	Feb 23, 2007 10:00:08 PM GMT+07:00
○	ADM_WEB	ANET_SEAT	REGNO	ADM_WEB	ANET_SEAT_IDX2	TABLE	ADM_WEB	NO	Feb 23, 2007 10:00:09 PM GMT+07:00
○	ADM_WEB	ANET_SUBJ_ADM52	REGNO, PPLID	ADM_WEB	ANET_SUBJ51_IDX1	TABLE	ADM_WEB	NO	Jan 16, 2009 10:01:21 PM GMT+07:00
○	ADM_WEB	ANET_GPAPR	PPLID, SYEAR, SCHOOLID, STUDID13	ADM_WEB	ANET_SUBJ51_IDX2	TABLE	ADM_WEB	NO	Dec 24, 2008 10:10:13 PM GMT+07:00
○	ADM_WEB	ANET_SUBJ_ADM52	PPLID, NAME	ADM_WEB	ANET_SUBJ51_IDX3	TABLE	ADM_WEB	NO	Jan 16, 2009 10:01:24 PM GMT+07:00
○	ADM_WEB	CEALSQUALANNO_ADM54	APP_ID, PPLID	ADM_WEB	ANN54_1	TABLE	ADM_WEB	NO	May 4, 2011 4:04:42 PM GMT+07:00
○	ADM_WEB	ANNOUNCEMENT50	APP_ID, PPLID	ADM_WEB	ANO_IDX1	TABLE	ADM_WEB	NO	May 11, 2007 10:00:10 PM GMT+07:00
○	ADM_WEB	ADM51_ANNOUNCE	APP_ID	ADM_WEB	APPID_IDX1	TABLE	ADM_WEB	NO	May 5, 2008 10:01:35 PM GMT+07:00
○	ADM_WEB	ANET_EXAMSEAT51	REGNO	ADM_WEB	A_SEAT51_IDX1	TABLE	ADM_WEB	NO	Feb 11, 2008 10:02:28 PM GMT+07:00
○	ADM_WEB	ANET_EXAMSEAT51	PPLID	ADM_WEB	A_SEAT51_IDX2	TABLE	ADM_WEB	NO	Feb 11, 2008 10:02:33 PM GMT+07:00
○	ADM_WEB	ANET_EXAMSEAT51	NAME	ADM_WEB	A_SEAT51_IDX3	TABLE	ADM_WEB	NO	Feb 11, 2008 10:02:34 PM GMT+07:00
○	ADM_WEB	ANET_EXAMSEAT51	SEATNO	ADM_WEB	A_SEAT51_IDX4	TABLE	ADM_WEB	NO	Feb 11, 2008 10:02:38 PM GMT+07:00

ภาพที่ 3.7 แสดงรายการดรรชนีบน Enterprise Manager ของระบบปัจจุบัน

จากภาพที่ 3.7 แสดงรายการดรรชนีจำนวนมากของระบบการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษาในปัจจุบัน ซึ่งมีจำนวนดรรชนีทั้งหมด 2514 ดรรชนี แต่ที่ใช้งานจริงมีเพียง 188 ดรรชนีเท่านั้น

หลักการจัดการดรรชนี

ดรรชนีช่วยให้การเข้าถึงข้อมูลในตารางสามารถทำได้เร็วยิ่งขึ้นกว่าการอ่านข้อมูลแบบธรรมดา (ไม่ใช้ดรรชนี) ดังนั้น ในการสร้างดรรชนีให้มีประสิทธิภาพ ผู้ดูแลฐานข้อมูลควรจะมีหลักการจัดการที่ถูกต้อง เพื่อที่จะสามารถใช้งานดรรชนีได้อย่างมีประสิทธิภาพดีที่สุด หลักการจัดการดรรชนี ประกอบด้วย

1) เลือกเขตข้อมูลที่ใช้ในการสร้างดรรชนีให้ถูกต้อง เมื่อเราทำงานกับข้อมูลโดยมีเงื่อนไขในการเลือกค่าข้อมูล (Where condition) เพื่อที่จะให้การทำงานนั้นรวดเร็วขึ้นเราควรจะนำเขตข้อมูลที่เป็นเงื่อนไขมาสร้างเป็นดรรชนี หากสร้างดรรชนีโดยใช้เขตข้อมูลอื่นที่ไม่ได้เป็นเงื่อนไขในการอ่านข้อมูลแล้วจะทำให้การอ่านข้อมูลช้า
2) สร้างดรรชนีให้อยู่คนละ Tablespace กับตาราง เพื่อเพิ่มประสิทธิภาพในการทำงานให้ดีมากยิ่งขึ้น ควรจะแยก Tablespace ที่เป็นดรรชนี และตาราง ออกจากกันเพื่อลดการแย่งใช้งานของพื้นที่ที่ใช้เก็บข้อมูลใน Tablespace นั้นๆ
3) กำนหดค่าการจองพื้นที่การใช้งานของดรรชนีทุกครั้ง ในการสร้างดรรชนีแต่ละครั้งควรจะประมาณขนาดของดรรชนีที่จะสร้างและความถี่ในการเปลี่ยนแปลงข้อมูลในตารางข้อมูลก่อนเสมอเพื่อที่จะสามารถจองพื้นที่การใช้งานของดรรชนีได้อย่างถูกต้อง
4) กำหนดจำนวนดรรชนีในแต่ละตารางข้อมูลไม่ให้มีมากเกินไป สามารถสร้างดรรชนีกี่ตัวก็ได้ใน 1 ตาราง แต่ถ้าจำนวนดรรชนีต่อ 1 ตารางมีมากเกินไปจะทำให้การสืบค้นข้อมูลในตารางนั้นช้าลง เนื่องจากหากมีการเปลี่ยนแปลงข้อมูลในเขตข้อมูลที่เป็นดรรชนี จะต้องมีการเปลี่ยนแปลงโครงสร้างของดรรชนีทุกตัวที่สร้างอยู่บนเขตข้อมูลนั้นทุกครั้งโดยอัตโนมัติ ซึ่งจะทำให้การทำงานช้าลง
5) สร้างดรรชนีหลังจากเพิ่มข้อมูลลงในตารางแล้ว ถ้าสร้างก่อนที่จะเพิ่มข้อมูลในตารางจะทำให้การทำงานนั้นช้าลง เนื่องจากต้องมีการเปลี่ยนแปลงโครงสร้างของดรรชนีทุกครั้ง ดังนั้น การเพิ่มข้อมูลจำนวนมากๆ ลงไปในตาราง ควรที่จะลบดรรชนีที่สร้างอยู่ในตารางนั้นออกก่อน และสร้างขึ้นมาใหม่หลังจากที่ทำงานเสร็จ
6) ลบดรรชนีที่ไม่ได้ใช้งาน ยิ่งจำนวนดรรชนีในตารางมีมากเท่าใดการทำงานยิ่งช้าลงเท่านั้น ดังนั้น ควรลบดรรชนีที่ไม่ได้ใช้งานออก
7) ควรสร้างดรรชนีใหม่ (Rebuild Index) อยู่เสมอ เมื่อโครงสร้างของดรรชนีถูกเปลี่ยนแปลงไปบ่อยๆ เนื่องจากข้อมูลมีการเปลี่ยนแปลงอยู่ตลอดเวลา จะทำให้

โครงสร้างของดรรชนีไม่สมดุล (Balance) ดังนั้น ควรสร้างดรรชนีใหม่บ่อยๆ เพื่อให้โครงสร้างของดรรชนีดีอยู่เสมอและประสิทธิภาพการใช้งานก็จะดีตามด้วย

การจัดการกับดรรชนีอย่างถูกวิธีจำเป็นอย่างยิ่งสำหรับการทำงานกับข้อมูลจำนวนมากและต้องการความรวดเร็วในการเข้าถึงข้อมูล ดรรชนีในฐานข้อมูลออราเคิลมีหลายประเภทและหลายรูปแบบ ควรเลือกสร้างดรรชนีให้เหมาะสมกับการใช้งานเพื่อให้ได้ประสิทธิภาพในการทำงานดีที่สุด

วิธีการสร้างดรรชนีให้มีประสิทธิภาพเร็วกว่าปกติ

การสร้างดรรชนีมีหลายแบบ [8] ได้แก่ B-Tree Index, Reversed Key Index, Bitmap Index และ Function-Based Index การสร้างจะใช้คำสั่งเดียวกันแต่จะแตกต่างกันเพียงตัวเลือกที่ต้องเลือกกำหนดในการสร้างดรรชนีแต่ละแบบ การสร้างดรรชนีสามารถสร้างได้บนเขตข้อมูลตั้งแต่ 1 เขตข้อมูลเป็นต้นไป โดยกำหนดให้ดรรชนีนั้นเก็บข้อมูลแบบ Unique Index หรือ Non-Unique Index ก็ได้

จากการวิเคราะห์การสร้างดรรชนีในระบบปัจจุบันทั้งหมด 2514 ดรรชนี ดังแสดงในภาพที่ 3.8 ดังนี้

INDEX_NAME	INDEX_TYPE	TABLE_NAME	TABLE_OWNER	UNIQUENESS	STATUS	FUN(
REGNO_INDX	NORMAL	SPC_SUBJ_ADM	ADM_WEB	NONUNIQUE	VALID	Null
PPLID_INDX	NORMAL	SPC_SUBJ_ADM	ADM_WEB	NONUNIQUE	VALID	Null
ONET_REGNO_INDX	NORMAL	ONET_SUBJ_ADM	ADM_WEB	NONUNIQUE	VALID	Null
CENTER_IDX	NORMAL	CENTER	ADM_WEB	NONUNIQUE	VALID	Null
ONET_PPLID_INDX	NORMAL	ONET_SUBJ_ADM	ADM_WEB	NONUNIQUE	VALID	Null
FIELD_IDX3	NORMAL	FIELD	ADM_WEB	NONUNIQUE	VALID	Null
FIELD_IDX2	NORMAL	FIELD	ADM_WEB	NONUNIQUE	VALID	Null
FIELD_IDX1	NORMAL	FIELD	ADM_WEB	NONUNIQUE	VALID	Null
ANET_CENTER_IDX	NORMAL	ANET_CENTER	ADM_WEB	NONUNIQUE	VALID	Null
ANET_FIELD_IDX1	NORMAL	ANET_FIELD	ADM_WEB	NONUNIQUE	VALID	Null
ANET_FIELD_IDX2	NORMAL	ANET_FIELD	ADM_WEB	NONUNIQUE	VALID	Null
ANET_FIELD_IDX3	NORMAL	ANET_FIELD	ADM_WEB	NONUNIQUE	VALID	Null
ONET_APP_P_ALL_IDX1	NORMAL	ONET_APP_P_ALL	ADM_WEB	NONUNIQUE	VALID	Null
ONET_APP_P_ALL_IDX2	NORMAL	ONET_APP_P_ALL	ADM_WEB	NONUNIQUE	VALID	Null
ONET_APP_P_ALL_IDX3	NORMAL	ONET_APP_P_ALL	ADM_WEB	NONUNIQUE	VALID	Null
ONET_APP_01_IDX1	NORMAL	ONET_APP_01	ADM_WEB	NONUNIQUE	VALID	Null
ONET_APP_01_IDX2	NORMAL	ONET_APP_01	ADM_WEB	NONUNIQUE	VALID	Null

ภาพที่ 3.8 แสดงรายการดรรชนีที่ใช้ระบบปัจจุบัน

จากภาพที่ 3.8 จากการวิเคราะห์การสร้างดรรชนีในระบบพบว่าทั้งหมดเป็นการสร้างดรรชนีแบบปกติ (Normal) ทุกดรรชนีไม่มีการกำหนดชนิดของดรรชนี ดังนั้น ผู้วิจัยจึงได้ทดลองปรับปรุงประสิทธิภาพของดรรชนีด้วยวิธีการกำหนดรูปแบบของดรรชนีเปรียบเทียบกับการสร้างดรรชนีแบบปกติ

ดรรชนีปกติคำสั่งที่ใช้ในการสร้างดรรชนี คือ

```
create index thesis_idx02 on
thesis_tb1(pplid);
```

และคำสั่งที่ใช้เปรียบเทียบ คือ

```
create bitmap index thesis_idx03 on
thesis_tb1(pplid);
```

และ

```
create unique index thesis_idx01 on
thesis_tb1(pplid) reverse;
```

โดยกำหนดให้สร้างดรรชนีแบบ Reversed Key Index ซึ่งจะกำหนดให้มีการกลับค่าข้อมูลก่อนสร้าง จากคำสั่งเอสคิวแอลต่อไปนี้

```
select thesis_tb1.pplid,count(*)
from thesis_tb1,thesis_tb2
where thesis_tb1.pplid = thesis_tb2.pplid
group by thesis_tb1.pplid
order by thesis_tb1.pplid;
```

ผลการประเมินประสิทธิภาพของการใช้ดรรชนี แสดงดังตารางที่ 3.1 ถึง ตารางที่ 3.4 ดังนี้

ตารางที่ 3.1 ผลการประเมินประสิทธิภาพการไม่ใช้ดรรชนี

จำนวนแถว	ขนาด (ไบต์)	ไม่ใช้ดรรชนี			
		Cost	%CPU	เวลา(นาที)	เวลา(วินาที)
3,212,694	0.5G	21,120	1	00:08:27	507
6,425,388	1G	64,501	1	00:12:54	774
9,638,082	1.5G	84,565	1	00:16:56	1,016
12,850,776	2G	105,000	1	00:21:03	1,263
16,063,470	2.5G	169,000	1	00:34:00	2,040
25,701,552	4.5G	339,000	1	01:07:54	4,074
51,403,104	7.5G	505,000	1	01:41:05	6,065
77,104,656	14G	1,012,000	1	03:22:31	12,151
154,209,312	21G	1,518,000	1	05:03:44	18,224
231,313,968	41G	2,530,000	1	08:26:05	30,365

ตารางที่ 3.2 ผลการประเมินประสิทธิภาพการใช้ดรรชนีแบบปกติ

จำนวนแถว	ขนาด (ไบต์)	ใช้ดรรชนีแบบปกติ			
		Cost	%CPU	เวลา(นาที)	เวลา(วินาที)
3,212,694	0.5G	19,092	2	00:03:50	230
6,425,388	1G	38,196	2	00:07:39	459
9,638,082	1.5G	33,725	1	00:06:45	405
12,850,776	2G	44,966	1	00:09:00	540
16,063,470	2.5G	92,537	3	00:18:31	1,111
25,701,552	4.5G	152,000	2	00:30:27	1,827
51,403,104	7.5G	179,000	1	00:35:59	2,159
77,104,656	14G	269,000	1	00:53:58	3,238
154,209,312	21G	539,000	1	01:47:55	6,475
231,313,968	41G	809,000	1	02:41:53	9,713

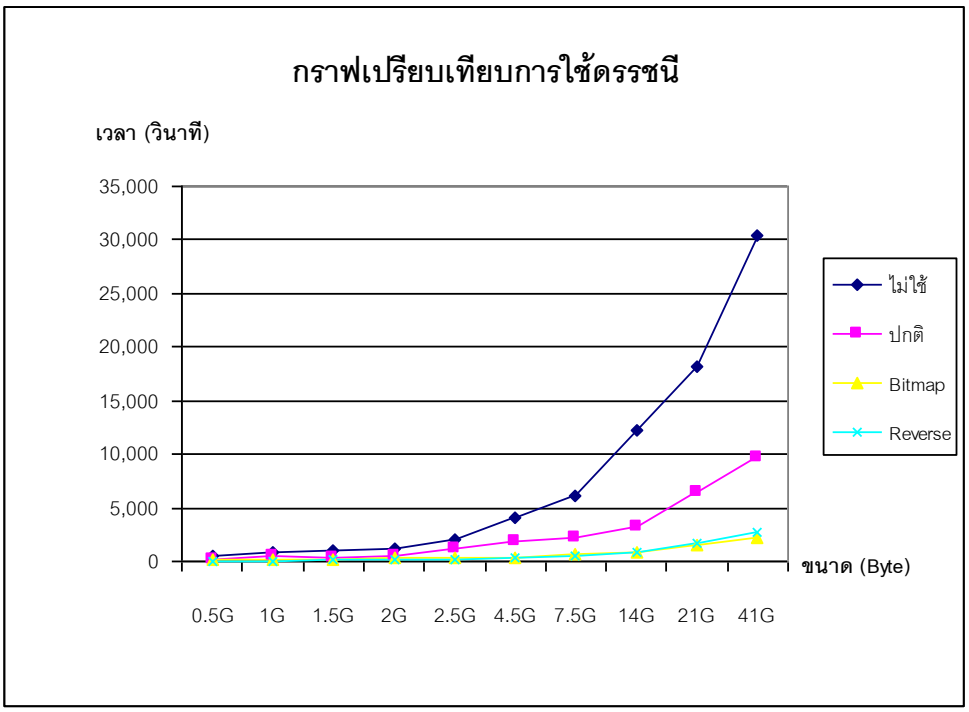
ตารางที่ 3.3 ผลการประเมินประสิทธิภาพการใช้ดรรชนี Bitmap

จำนวนแถว	ขนาด (ไบต์)	ใช้ดรรชนีแบบ Bitmap			
		Cost	%CPU	เวลา(นาที)	เวลา(วินาที)
3,212,694	0.5G	13,045	1	00:02:37	157
6,425,388	1G	17,184	1	00:03:27	207
9,638,082	1.5G	19,814	1	00:03:58	238
12,850,776	2G	22,399	1	00:04:29	269
16,063,470	2.5G	24,633	1	00:04:56	296
25,701,552	4.5G	28,923	1	00:05:48	348
51,403,104	7.5G	53,159	1	00:10:38	638
77,104,656	14G	73,066	1	00:14:37	877
154,209,312	21G	130,000	1	00:26:05	1,565
231,313,968	41G	186,000	1	00:37:20	2,240

ตารางที่ 3.4 ผลการประเมินประสิทธิภาพการใช้ดรรชนีแบบ Reverse

จำนวนแถว	ขนาด (ไบต์)	ใช้ดรรชนีแบบ Reverse			
		Cost	%CPU	เวลา(นาที)	เวลา(วินาที)
3,212,694	0.5G	3,067	2	00:00:37	37
6,425,388	1G	6,131	2	00:01:14	74
9,638,082	1.5G	9,196	2	00:01:51	111
12,850,776	2G	12,261	2	00:02:28	148
16,063,470	2.5G	15,325	2	00:03:04	184
25,701,552	4.5G	24,712	2	00:04:57	297
51,403,104	7.5G	49,037	2	00:09:49	589
77,104,656	14G	73,554	2	00:14:43	883
154,209,312	21G	147,000	2	00:29:26	1,766
231,313,968	41G	220,000	2	00:44:08	2,648

จากตารางที่ 3.1 ถึง ตารางที่ 3.4 สามารถนำมาแสดงเปรียบเทียบเป็นกราฟได้ดังภาพที่ 3.9 ดังนี้



ภาพที่ 3.9 กราฟเปรียบเทียบการใช้ดรรชนี

จากตารางที่ 3.1 ถึง ตารางที่ 3.4 และภาพที่ 3.9 จะเห็นว่าการใช้ดรรชนีช่วยให้การสืบค้นข้อมูลจากฐานข้อมูลรวดเร็วขึ้น ทั้งแบบปกติ แบบ Bitmap และ Revered Key สามารถช่วยให้การสืบค้นข้อมูลเร็วขึ้น และจากกราฟจะเห็นว่าการสร้างดรรชนีแบบ Reversed Key ใช้เวลาในการประมวลผลน้อยที่สุด จึงเหมาะสมสำหรับคำสั่งเอสคิวแอลดังกล่าวและข้อมูลขนาดใหญ่ เพราะฉะนั้นผู้ใช้ควรเลือกรูปแบบการสร้างดรรชนีที่เหมาะสมที่ได้ผลลัพธ์ออกมาโดยใช้เวลาน้อยที่สุด การสร้างดรรชนีที่เหมาะสมกับการใช้งานจะช่วยให้ประสิทธิภาพในการทำงานดีขึ้น

แนวทางที่ 2 การปรับปรุงประสิทธิภาพฐานข้อมูลของระบบโดยการใช้ Parallel Hint[16] Parallel เป็น Hint ตัวหนึ่งที่ช่วยในการจัดการกับข้อมูลขนาดใหญ่เพื่อเพิ่มประสิทธิภาพในฐานข้อมูล รูปแบบคำสั่ง hint คือ /*+PARALLEL(table_name degree)*/

ตัวอย่างการปรับปรุงประสิทธิภาพฐานข้อมูลด้วย Parallel Hint แสดงดังภาพที่ 3.10 ดังนี้

```
SQL> select /*+ parallel(thesis_gpa1) */ thesis_gpa1.pplid,count(*)
  2  from thesis_gpa1,thesis_gpa2
  3  where thesis_gpa1.pplid = thesis_gpa2.pplid
  4  group by thesis_gpa1.pplid
  5  order by thesis_gpa1.pplid;

Execution Plan
----------------------------------------------------------
Plan hash value: 3803233412

-------------------------------------------------------------------------------------------------------------------------------
| Id  | Operation                  | Name        | Rows  | Bytes |TempSpc| Cost (%CPU)| Time     |    TQ  |IN-OUT| PQ Distrib |
-------------------------------------------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT           |             |  3005K|    80M|       | 40146   (1)| 00:08:02 |        |      |            |
|   1 |  PX COORDINATOR            |             |       |       |       |            |          |        |      |            |
|   2 |   PX SEND QC (ORDER)       | :TQ10003    |  3005K|    80M|       | 40146   (1)| 00:08:02 |  Q1,03 | P->S | QC (ORDER) |
|   3 |    SORT ORDER BY           |             |  3005K|    80M|   237M| 40146   (1)| 00:08:02 |  Q1,03 | PCWP |            |
|   4 |     PX RECEIVE             |             |  3005K|    80M|       | 40146   (1)| 00:08:02 |  Q1,03 | PCWP |            |
|   5 |      PX SEND RANGE         | :TQ10002    |  3005K|    80M|       | 40146   (1)| 00:08:02 |  Q1,02 | P->P | RANGE      |
|   6 |       SORT GROUP BY        |             |  3005K|    80M|   237M| 40146   (1)| 00:08:02 |  Q1,02 | PCWP |            |
|*  7 |        HASH JOIN           |             |  6858K|   183M|       | 40144   (1)| 00:08:02 |  Q1,02 | PCWP |            |
|   8 |         PX RECEIVE         |             |  3212K|    42M|       |    43   (3)| 00:00:01 |  Q1,02 | PCWP |            |
|   9 |          PX SEND HASH      | :TQ10001    |  3212K|    42M|       |    43   (3)| 00:00:01 |  Q1,01 | P->P | HASH       |
|  10 |           PX BLOCK ITERATOR|             |  3212K|    42M|       |    43   (3)| 00:00:01 |  Q1,01 | PCWC |            |
|  11 |            TABLE ACCESS FULL| THESIS_GPA1 |  3212K|    42M|       |    43   (3)| 00:00:01 |  Q1,01 | PCWP |            |
|  12 |         BUFFER SORT        |             |       |       |       |            |          |  Q1,02 | PCWC |            |
|  13 |          PX RECEIVE        |             |  6425K|    85M|       | 40101   (1)| 00:08:02 |  Q1,02 | PCWP |            |
|  14 |           PX SEND HASH     | :TQ10000    |  6425K|    85M|       | 40101   (1)| 00:08:02 |        | S->P | HASH       |
|  15 |            TABLE ACCESS FULL| THESIS_GPA2 |  6425K|    85M|       | 40101   (1)| 00:08:02 |        |      |            |
-------------------------------------------------------------------------------------------------------------------------------
```

ภาพที่ 3.10 แสดงแผนการสืบค้นด้วยการใช้ Parallel Hint

แผนการสืบค้นก่อนการใช้ Parallel Hint แสดงดังรูปที่ 3.11 ดังนี้

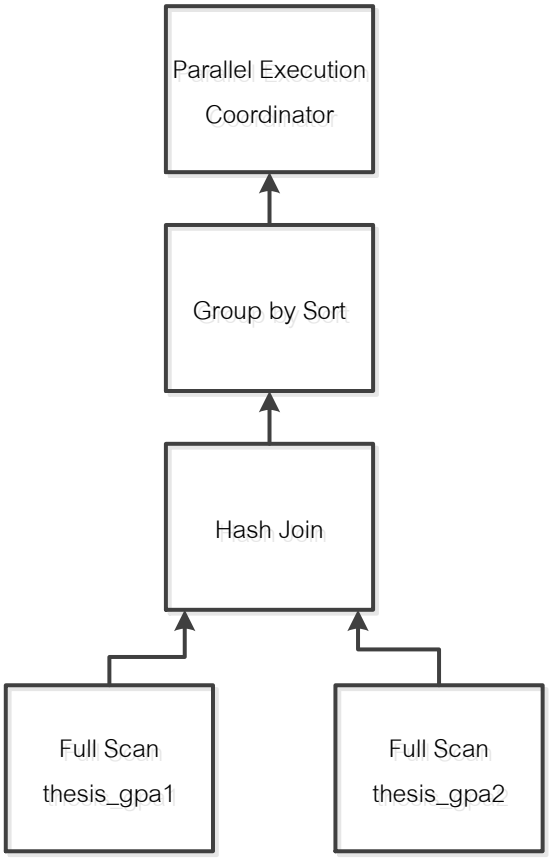
```
SQL> select thesis_gpa1.pplid,count(*)
  2  from thesis_gpa1,thesis_gpa2
  3  where thesis_gpa1.pplid = thesis_gpa2.pplid
  4  group by thesis_gpa1.pplid
  5  order by thesis_gpa1.pplid;

Execution Plan
----------------------------------------------------------
Plan hash value: 2543678536

-------------------------------------------------------------------------------------------------
| Id  | Operation              | Name        | Rows  | Bytes |TempSpc| Cost (%CPU)| Time     |
-------------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT       |             |  3005K|   117M|       | 96978   (2)| 00:19:24 |
|   1 |  SORT GROUP BY NOSORT  |             |  3005K|   117M|       | 96978   (2)| 00:19:24 |
|   2 |   MERGE JOIN           |             |  3208K|   125M|       | 96978   (2)| 00:19:24 |
|   3 |    SORT JOIN           |             |  3005K|    77M|       | 61271   (2)| 00:12:16 |
|   4 |     VIEW               | VW_GBF_7    |  3005K|    77M|       | 61271   (2)| 00:12:16 |
|   5 |      HASH GROUP BY     |             |  3005K|    40M|   123M| 61271   (2)| 00:12:16 |
|   6 |       TABLE ACCESS FULL| THESIS_GPA2 |  6425K|    85M|       | 40101   (1)| 00:08:02 |
|*  7 |    SORT JOIN           |             |  3212K|    42M|   123M| 35707   (2)| 00:07:09 |
|   8 |     TABLE ACCESS FULL  | THESIS_GPA1 |  3212K|    42M|       | 19683   (1)| 00:03:57 |
-------------------------------------------------------------------------------------------------
```

ภาพที่ 3.11 แสดงแผนการสืบค้นโดยไม่ใช้ Parallel Hint

จากภาพที่ 3.10 และ 3.11 จะเห็นว่าการปรับปรุงโดยใช้ Parallel Hint จะช่วยให้การสืบค้นเร็วขึ้นจาก 00:03:57 วินาที เหลือ 00:00:01 วินาที เร็วขึ้นถึงร้อยละ 99.5 ซึ่งแผนภาพการไหลของข้อมูล (Data Flow Diagram) สำหรับการ Join ตารางของคำสั่งเอสคิวแอล

```
select /*+ parallel(thesis_gpa1 2) */ thesis_gpa1.pplid,count(*)
from thesis_gpa1,thesis_gpa2
where thesis_gpa1.pplid = thesis_gpa2.pplid
group by thesis_gpa1.pplid
order by thesis_gpa1.pplid;
```

แผนภาพการไหลของข้อมูลแสดงดังภาพที่ 3.12 ดังนี้



ภาพที่ 3.12 แผนภาพการไหลของข้อมูลสำหรับการ Join ตาราง [18]

จากภาพที่ 3.12 แสดงแผนภาพการไหลของข้อมูลการ Join ตาราง ประกอบด้วยตาราง thesis_gpa1 และ thesis_gpa2 เริ่มจากการสแกนตาราง thesis_gpa1 และ thesis_gpa2 ใน Parallel จากนั้นจะส่งจำนวนแถวไปทำการ Hash Join และ ตามด้วย Group by ตามลำดับ เพื่อประมวลผลคำสั่งดังเอสคิวแอลดังกล่าว สำหรับงานวิจัยนี้ได้ทดลองการใช้ Parallel Hint ในระดับต่างๆ เพื่อวิเคราะห์ประสิทธิภาพของการประมวลผลคำสั่งเอสคิวแอลโดยการใช้ Parallel Hint ในขนาดข้อมูลจำนวนมาก ซึ่งจากการทดลองเปรียบเทียบการใช้และไม่ใช้ Parallell Hint แสดงดังตารางที่ 3.2 ดังนี้

ตารางที่ 3.5 เปรียบเทียบการใช้ Parallel Hint และไม่ใช้ Parallel Hint

จำนวนข้อมูล (แถว)	**ขนาด (Byte)**	**ผลลัพธ์**	**ไม่ใช้ Parallel Hint (วินาที)**	**ใช้ Parallel Hint (วินาที)**	**อัตราลดลง (ร้อยละ)**	**ระดับ (Degree)**
3,212,694	0.5G	✓	00:03:57	00:03:57	0.00	1
3,212,694	0.5G	✓	00:03:57	00:02:11	44.73	2
3,212,694	0.5G	✓	00:03:57	00:01:28	62.87	3
3,212,694	0.5G	✓	00:03:57	00:01:06	72.15	4
3,212,694	0.5G	✓	00:03:57	00:00:53	77.64	5
3,212,694	0.5G	✓	00:03:57	00:00:44	81.43	6
3,212,694	0.5G	✓	00:03:57	00:00:38	83.97	7
3,212,694	0.5G	✓	00:03:57	00:00:33	86.08	8
3,212,694	0.5G	✓	00:03:57	00:00:30	87.34	9
3,212,694	0.5G	✓	00:03:57	00:00:27	88.61	10
6,425,388	1G	✓	00:08:02	00:08:02	0.00	1
6,425,388	1G	✓	00:08:02	00:04:27	44.61	2
6,425,388	1G	✓	00:08:02	00:02:58	63.07	3
6,425,388	1G	✓	00:08:02	00:02:14	72.20	4
6,425,388	1G	✓	00:08:02	00:01:47	77.80	5
6,425,388	1G	✓	00:08:02	00:01:29	81.54	6
6,425,388	1G	✓	00:08:02	00:01:17	84.02	7
6,425,388	1G	✓	00:08:02	00:01:07	86.10	8
6,425,388	1G	✓	00:08:02	00:01:00	87.55	9
6,425,388	1G	✓	00:08:02	00:00:54	88.80	10
9,638,082	1.5G	✓	00:12:03	00:12:03	0.00	1
9,638,082	1.5G	✓	00:12:03	00:06:41	44.54	2
9,638,082	1.5G	✓	00:12:03	00:04:28	62.93	3
9,638,082	1.5G	✓	00:12:03	00:03:21	72.20	4

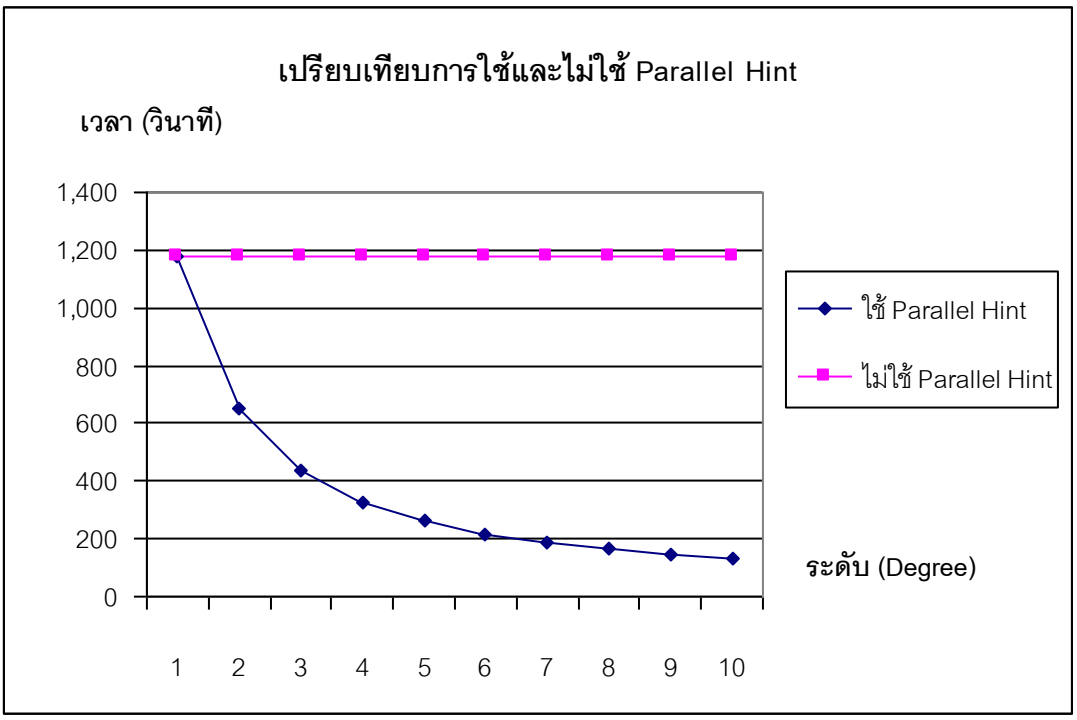
จำนวนข้อมูล (แถว)	ขนาด (Byte)	ผลลัพธ์	ไม่ใช้ Parallel Hint (วินาที)	ใช้ Parallel Hint (วินาที)	อัตราลดลง (ร้อยละ)	ระดับ (Degree)
9,638,082	1.5G	✓	00:12:03	00:02:41	77.73	5
9,638,082	1.5G	✓	00:12:03	00:02:14	81.47	6
9,638,082	1.5G	✓	00:12:03	00:01:55	84.09	7
9,638,082	1.5G	✓	00:12:03	00:01:41	86.03	8
9,638,082	1.5G	✓	00:12:03	00:01:30	87.55	9
9,638,082	1.5G	✓	00:12:03	00:01:21	88.80	10
12,850,776	2G	✓	00:16:04	00:16:04	0.00	1
12,850,776	2G	✓	00:16:04	00:08:55	44.50	2
12,850,776	2G	✓	00:16:04	00:05:57	62.97	3
12,850,776	2G	✓	00:16:04	00:04:28	72.20	4
12,850,776	2G	✓	00:16:04	00:03:34	77.80	5
12,850,776	2G	✓	00:16:04	00:02:59	81.43	6
12,850,776	2G	✓	00:16:04	00:02:33	84.13	7
12,850,776	2G	✓	00:16:04	00:02:14	86.10	8
12,850,776	2G	✓	00:16:04	00:01:59	88.59	9
12,850,776	2G	✓	00:16:04	00:01:47	88.90	10
16,063,470	2.5G	✓	00:19:39	00:19:39	0.00	1
16,063,470	2.5G	✓	00:19:39	00:10:54	44.53	2
16,063,470	2.5G	✓	00:19:39	00:07:16	63.02	3
16,063,470	2.5G	✓	00:19:39	00:05:27	72.26	4
16,063,470	2.5G	✓	00:19:39	00:04:22	77.78	5
16,063,470	2.5G	✓	00:19:39	00:03:38	81.51	6
16,063,470	2.5G	✓	00:19:39	00:03:07	84.14	7
16,063,470	2.5G	✓	00:19:39	00:02:44	86.09	8
16,063,470	2.5G	✓	00:19:39	00:02:26	87.62	9

จำนวนข้อมูล (แถว)	ขนาด (Byte)	ผลลัพธ์	ไม่ใช้ Parallel Hint (วินาที)	ใช้ Parallel Hint (วินาที)	อัตราลดลง (ร้อยละ)	ระดับ (Degree)
16,063,470	2.5G	✓	00:19:39	00:02:11	88.89	10
25,701,552	4.5G	✓	00:31:41	00:31:41	0.00	1
25,701,552	4.5G	✓	00:31:41	00:17:34	44.56	2
25,701,552	4.5G	✓	00:31:41	00:11:43	63.02	3
25,701,552	4.5G	✓	00:31:41	00:08:47	72.28	4
25,701,552	4.5G	✓	00:31:41	00:07:02	77.80	5
25,701,552	4.5G	✓	00:31:41	00:05:52	81.48	6
25,701,552	4.5G	✓	00:31:41	00:05:02	84.11	7
25,701,552	4.5G	✓	00:31:41	00:04:24	86.11	8
25,701,552	4.5G	✓	00:31:41	00:03:55	87.64	9
25,701,552	4.5G	✓	00:31:41	00:03:31	88.90	10

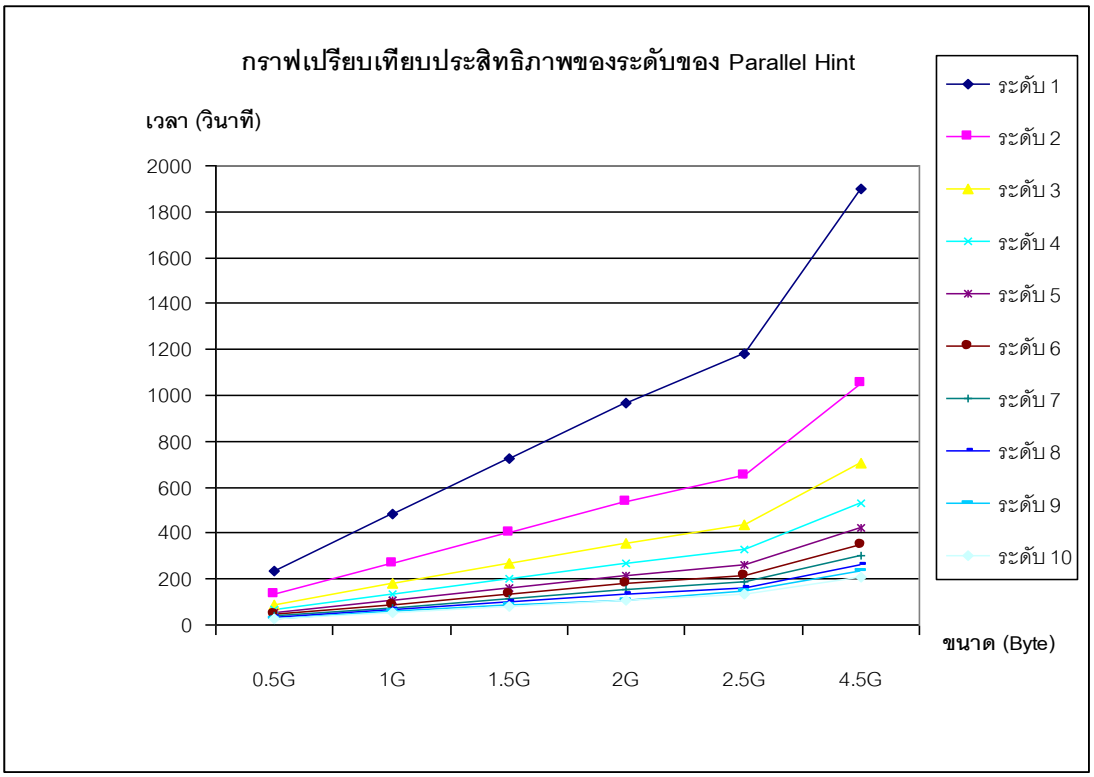
จากตารางที่ 3.5 จะเห็นว่าการใช้ Parallel Hint ช่วยให้การประมวลผลคำสั่งเอสแอลมีประสิทธิภาพเร็วขึ้น โดยที่ยิ่งระดับ (Degree) มากยิ่งทำให้ประสิทธิภาพดีขึ้น สามารถนำข้อมูลการใช้ Parallel Hint จากตารางที่ 3.5 นำมาแสดงเป็นกราฟดังภาพที่ 3.14 และนำข้อมูลจากตารางที่ 3.5 ซึ่งใช้คำสั่งการสืบค้นต่อไปนี้

```
select /*+ parallel(thesis_gpa5 1) */ thesis_gpa5.pplid,count(*)
from thesis_gpa5,thesis_gpa6
where thesis_gpa5.pplid = thesis_gpa6.pplid
group by thesis_gpa5.pplid
order by thesis_gpa5.pplid;
```

จากคำสั่งสืบค้นดังกล่าวมีข้อมูลจำนวน 16,063,470 แถว ขนาด 2.5 กิกะไบต์ นำมาแสดงเป็นกราฟได้ดังภาพที่ 3.13 ดังนี้



ภาพที่ 3.13 กราฟเปรียบเทียบการใช้และไม่ใช้ Parallel Hint



ภาพที่ 3.14 กราฟเปรียบเทียบประสิทธิภาพของระดับของ Parallel Hint

จากกราฟในภาพที่ 3.13 และ 3.14 แสดงการสืบค้นโดยการใช้และไม่ใช้ Parallel Hint และประสิทธิภาพของการใช้ Parallel Hint ในแต่ละระดับตั้งแต่ระดับ 1 ถึง ระดับ 10 ผลปรากฏว่า การใช้ Parallel Hint ระดับ 1 ใช้เวลาเท่ากับการไม่ใช้ Parallel Hint และการกำหนดใช้ระดับ 10 ใช้เวลาในการสืบค้นข้อมูลน้อยที่สุดเมื่อเทียบกับระดับ 1 ถึง ระดับ 9 ดังนั้น จะเห็นได้ว่ายิ่งระดับมากขึ้นเท่าไหร่การสืบค้นยิ่งใช้เวลาน้อยเท่านั้น

3.4 การพัฒนาเครื่องมือเพื่อวัดประสิทธิภาพ

การพัฒนาเครื่องมือวัดประสิทธิภาพของฐานข้อมูลได้ดำเนินการตามขั้นตอนดังต่อไปนี้

3.2.1 ศึกษาการเก็บข้อมูล และการบันทึกผลการทำงานของฐานข้อมูลออราเคิล

3.2.2 ออกแบบการทำงานของเครื่องมือ

3.2.3 การพัฒนาเครื่องมือ

3.2.4 การทดสอบเครื่องมือ

ซึ่งรายละเอียดจะกล่าวถึงในบทที่ 4

3.5 การทดสอบการปรับปรุงฐานข้อมูล

การทดสอบการปรับปรุงฐานข้อมูลตามแนวทางที่ออกแบบ สามารถทำได้ตามขั้นตอนต่อไปนี้

3.5.1 นำข้อมูลของระบบกรณีศึกษา ตั้งแต่ พ.ศ. 2552 ถึง พ.ศ. 2555 มาสร้างและออกแบบใหม่ตามแนวทางข้อ 3.3

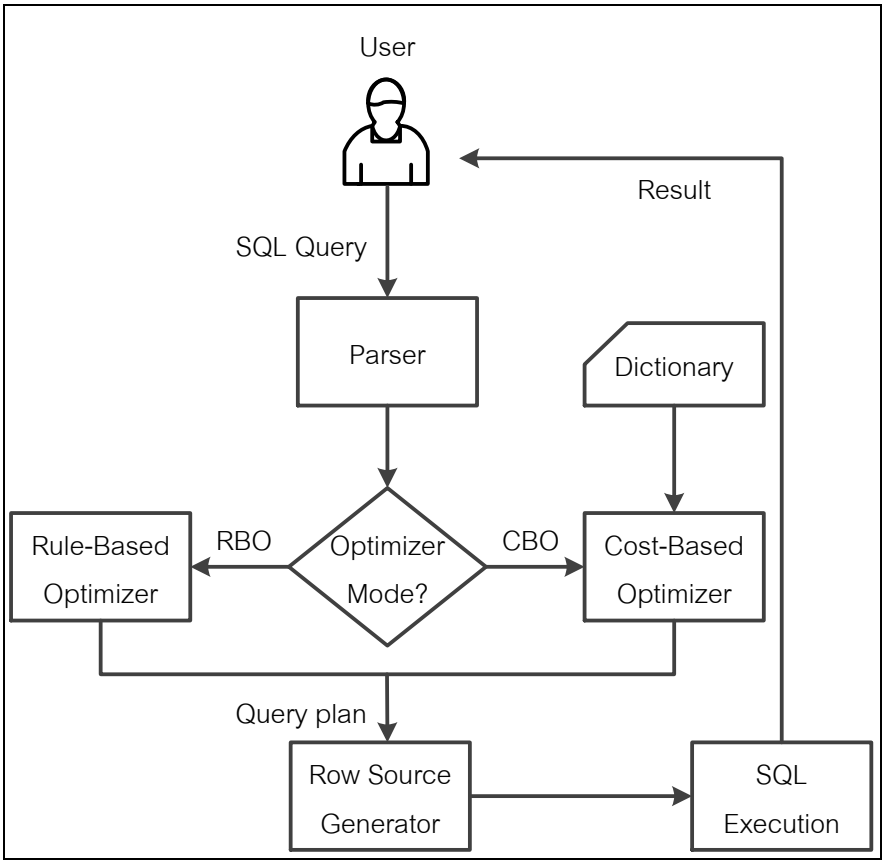
3.5.2 ทดสอบคำสั่งเอสคิวแอลแบบต่างๆ ที่ได้ผลลัพธ์ถูกต้องตามโปรแกรมประยุกต์ที่ใช้ในระบบ และวัดประสิทธิภาพการทำงานโดยใช้โปรแกรมสำเร็จรูปและเครื่องมือที่พัฒนาขึ้น

3.5.2.1 แนวคิดการประมวลผลคำสั่งเอสคิวแอล [10]

โดยทั่วไปผู้ใช้จะเปลี่ยนแปลงหรือแก้ไขข้อมูลต่างๆ ภายในฐานข้อมูลผ่านซอฟต์แวร์ที่เรียกว่า ระบบจัดการฐานข้อมูลหรือดีบีเอ็มเอส (DBMS: Database Management System) โดยใช้ภาษาเอสคิวแอล ซึ่งเป็นภาษามาตรฐานที่ถูกออกแบบมาเพื่อจัดการข้อมูลผ่านระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS)

คอมโพเน็นต์ในระบบจัดการฐานข้อมูลที่นับเป็นกลไกสำคัญในการจัดการงานดังกล่าวได้นั้นเรียกว่า อ๊อพติไมเซอร์ (Optimizer) สามารถจัดการและ

ประมวลผลคำสั่งสืบค้นและคำสั่งจัดการข้อมูลต่างๆ ได้ ซึ่งมีขั้นตอนในการประมวลผลคำสั่งเอสคิวแอลดังแสดงในภาพที่ 3.15 ดังนี้



ภาพที่ 3.15 แสดงการทำงานของอ๊อพติไมเซอร์ในการประมวลผลคำสั่ง [10]

จากภาพที่ 3.15 สามารถอธิบายได้ดังต่อไปนี้

1) เมื่อระบบได้รับคำสั่งเอสคิวแอลจากผู้ใช้ระบบจะส่งชุดคำสั่งดังกล่าวไปยังคอมโพเน็นต์ที่เรียกว่า Parser ให้ทำการวิเคราะห์รูปแบบและความถูกต้องของประโยค (Syntax Checking) รวมถึงตรวจสอบถึงความถูกต้องของอ็อบเจ็คหรือตารางต่างๆ ที่มีการอ้างถึงในระบบฐานข้อมูล ซึ่งผลลัพธ์จะได้รูปแบบคำสั่งเอสคิวแอลที่ผ่าน การตรวจสอบตามโครงสร้างที่ประโยคที่ถูกต้องแล้ว (Verified SQL Query)

2) จากนั้นจะส่งผลคำสั่งที่ผ่านการตรวจสอบในข้อ 1 ไปยังอ็อพติไมเซอร์เพื่อวิเคราะห์และสร้างแผนการสืบค้น โดยมี 2 รูปแบบคือแบบ RBO (Ruled Based Optimizer) ซึ่งใช้กฎความสัมพันธ์เพื่อสร้างการประมวลผลคำสั่งหรือ

สร้างดรรชนีให้กับข้อมูล ในขณะที่ CBO (Cost Based Optimizer) จะใช้ข้อมูลสถิติเข้ามาช่วย ประเมินและกำหนดแผนการสืบค้นก่อนประมวลผลหรือรันคำสั่งจริง เช่น จำนวนแถวข้อมูลหรือไบต์ข้อมูลที่เข้าถึง เป็นต้น

3) จากนั้นเมื่อ Row Source Generator ได้รับแผนการสืบค้นแล้วจะสร้างความสัมพันธ์และลำดับการทำงานในรูปแบบของทรี (Tree Structure) ก่อนส่งโครงสร้างดังกล่าวไปยัง SQL Executor เพื่อประมวลผลคำสั่งตามแผนดังกล่าวก่อนส่งผลลัพธ์ไปแสดงผลยังผู้ใช้ต่อไป

ตัวอย่างแผนการสืบค้น (Execution Plan) ของคำสั่งเอสคิวเอส คำสั่ง Select จำนวน 411,224,832 แถว (ขนาด 58 กิกะไบต์) ก่อนปรับปรุงแสดงดังภาพที่ 3.16 ดังนี้

```
SQL> select * from thesis_gpa;

Execution Plan
----------------------------------------------------------
Plan hash value: 3031678280

-------------------------------------------------------------------------------
| Id  | Operation         | Name       | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT  |            |   411M|    58G|  2941K  (2)| 09:48:23 |
|   1 |  TABLE ACCESS FULL| THESIS_GPA |   411M|    58G|  2941K  (2)| 09:48:23 |
-------------------------------------------------------------------------------
```

ภาพที่ 3.16 ตัวอย่างแผนการสืบค้นของคำสั่ง Select ก่อนปรับปรุง

และตัวอย่างแผนการสืบค้นหลังปรับปรุงแสดงดังภาพที่ 3.17 ดังนี้

```
SQL> select name from thesis_gpa;

Execution Plan
----------------------------------------------------------
Plan hash value: 3031678280

-------------------------------------------------------------------------------
| Id  | Operation         | Name       | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | SELECT STATEMENT  |            |   411M|    15G|  2915K  (1)| 09:43:09 |
|   1 |  TABLE ACCESS FULL| THESIS_GPA |   411M|    15G|  2915K  (1)| 09:43:09 |
-------------------------------------------------------------------------------
```

ภาพที่ 3.17 ตัวอย่างแผนการสืบค้นของคำสั่ง Select หลังปรับปรุง

แผนการสืบค้นประกอบด้วยข้อมูลพื้นฐานต่างๆ ดังนี้คือ

1. การดำเนินการ (Operation) เป็นรายการทำงานหรือคำสั่งย่อยของการดำเนินการต่างๆ ที่อ๊อพติไมเซอร์จะประมวลผล โดยจะเริ่มรันจากโหนดลูกหรือ Child Node ก่อน (โอเปอเรชันที่อยู่ด้านขวามือสุด) จากนั้นรันโหนดแม่

หรือ Parent Node (โอเปอเรชันที่อยู่ใกล้ที่สุดของโหนดลูกนั้น) แต่หากมีโอเปอเรชันที่ระดับเท่ากันก็ให้เริ่มรันที่รหัสด้านบนก่อน [10]

2. ชื่อวัตถุฐานข้อมูล (Name) เป็นชื่อวัตถุฐานข้อมูล (Database Object) ที่มีการอ้างถึง เช่น ตาราง หรือดรรชนีที่มีการใช้งาน
3. จำนวนแถวและไบต์ (Rows/Bytes) เป็นจำนวนแถวและไบต์ที่มีการอ่านหรือโหลดข้อมูลเข้าในระบบ
4. ค่าน้ำหนักที่สูญเสียไป (Cost) เป็นค่าน้ำหนักที่ระบบสูญเสียไปในการประมวลผลคำสั่งเอสคิวแอลนั้น ซึ่งจะไม่มีหน่วยการวัด โดยอ๊อพติไมเซอร์จะคำนวณค่าดังกล่าวจากการใช้งาน ซีพียู (CPU Cost) และค่าการติดต่อใช้งานกับดิสก์
5. ร้อยละของการใช้งานหน่วยประมวลผลกลาง (%CPU) เป็นค่าอัตราส่วนที่อ๊อพติไมเซอร์ใช้ ประมวลผลและ ทำงานสำหรับโอเปอเรชันนั้นๆ
6. เวลา (Time) เป็นเวลาที่ใช้ในการประมวลผลคำสั่งมีหน่วยเป็นวินาที (ms: Millisecond)

3.5.3 สรุปรูปแบบเอสคิวแอลที่เหมาะสมที่ทดสอบจากข้อ 3.5.2

3.6 การประเมินผลการปรับปรุงประสิทธิภาพ

การเลือกตัววัดประสิทธิภาพสำหรับการปรับปรุงประสิทธิภาพฐานข้อมูลของกรณีศึกษา ในการวิเคราะห์ประสิทธิภาพ ได้แก่ การเลือกเกณฑ์หรือหลักการในการเปรียบเทียบประสิทธิภาพ หลักการดังกล่าวเรียกว่า ตัววัดประสิทธิภาพ (Performance Metrics) ซึ่งตัววัดประสิทธิภาพของการประมวลผลของระบบ คือ เวลาในการประมวลผลของระบบ ซึ่งใช้เทคนิคการประเมินโดยการวัดจากระบบจริง และการวิเคราะห์ผลที่ได้จากการปรับปรุงประสิทธิภาพของระบบ

ซึ่งรายละเอียดผลของการประเมินการปรับปรุงประสิทธิภาพจะกล่าวถึงในบทที่ 5

บทที่ 4

การออกแบบและพัฒนาเครื่องมือ

การออกแบบและพัฒนาเครื่องมือในวิทยานิพนธ์นี้ เริ่มจากสภาพแวดล้อมที่ใช้สำหรับการพัฒนาเครื่องมือ จากนั้นอธิบายการออกแบบและโครงสร้างของเครื่องมือ และโครงสร้างฐานข้อมูลของเครื่องมือ ซึ่งมีรายละเอียดดังต่อไปนี้

4.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ

สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือมีรายละเอียด ดังนี้

1. ฮาร์ดแวร์ (Hardware)

1.1 เครื่องคอมพิวเตอร์แบบโน้ตบุ๊ก (Notebook) หน่วยประมวลผลอินเทลคอร์ไอไฟว์ 2.40 กิกะเฮิร์ท (Intel Core i5 CPU 2.40 GHz)

1.2 หน่วยความจำสำรอง (RAM) 4 กิกะไบต์ (4 GB)

1.3 ฮาร์ดดิสก์ (Harddisk) 300 กิกะไบต์ (300 GB)

2. ซอฟต์แวร์ (Software)

2.1 ระบบปฏิบัติการ(Operating System) ไมโครซอฟท์วินโดวส์เซเวน(Microsoft Windows 7)

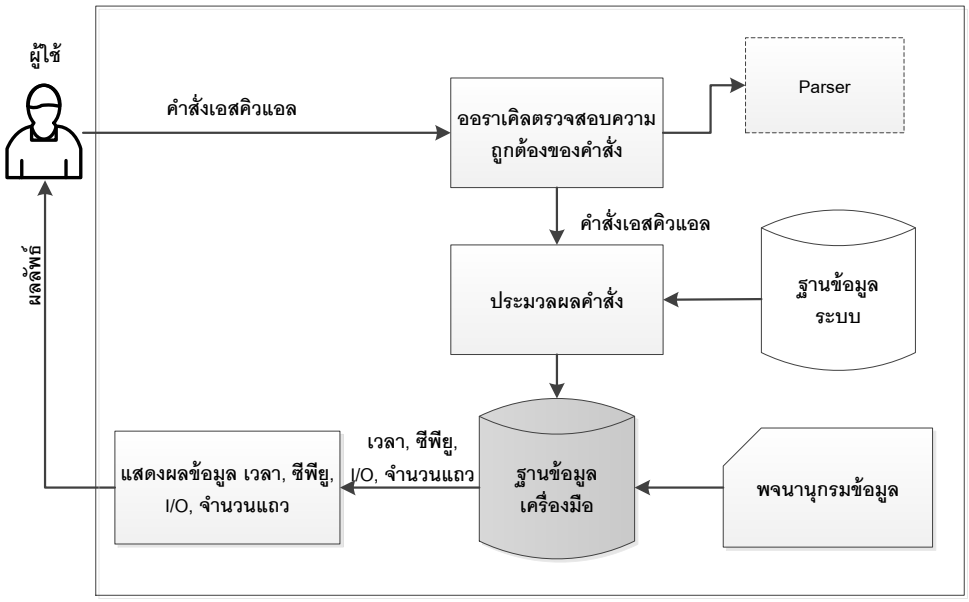
2.2 แมโครมีเดีย ดรีมวีฟเวอร์ เวอร์ชัน 8 (Macromedia Dreamweaver 8)

2.3 ภาษาพีเอชพี เวอร์ชัน 5.2.5 (PHP 5.2.5)

2.3 ฐานข้อมูลออราเคิลเวอร์ชัน 11 จี (Oracle 11g)

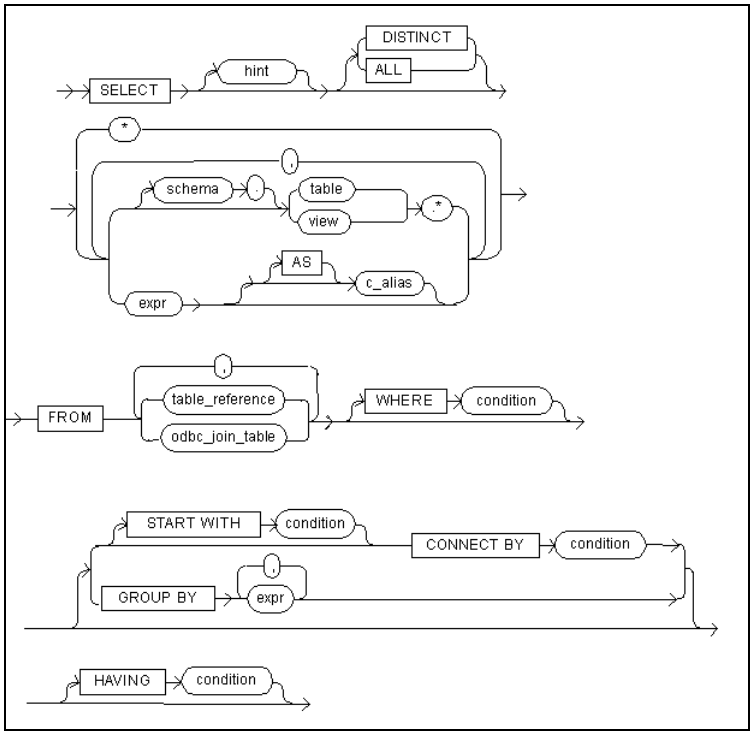
4.2 การออกแบบและโครงสร้างของเครื่องมือ

นอกจากการทำเสนอแนวทางการปรับปรุงประสิทธิภาพของระบบที่ใช้ฐานข้อมูลออราเคิลแล้ว วิทยานิพนธ์นี้ยังได้จัดทำเครื่องมือการวัดประสิทธิภาพของฐานข้อมูลโดยมีวัตถุประสงค์เพื่อแสดงผลประสิทธิภาพของคำสั่งเอสคิวแอลที่ใช้ในระบบ และเพื่อประโยชน์สำหรับนักพัฒนาในการเลือกใช้คำสั่งเอสคิวแอลที่เหมาะสม ซึ่งโครงสร้างการทำงานของเครื่องมือแสดงดังภาพที่ 4.1 ดังนี้



ภาพที่ 4.1 โครงสร้างการทำงานของเครื่องมือ

จากภาพที่ 4.1 แสดงโครงสร้างของเครื่องมือวัดประสิทธิภาพของฐานข้อมูล เริ่มจากผู้ใช้นำข้อมูลคำสั่งเอสคิวเข้าสู่ระบบ ออราเคิลจะทำการตรวจสอบความถูกต้องของคำสั่ง หลังจากตรวจสอบความถูกต้องของคำสั่งแล้วจะทำการประมวลผลคำสั่งตามขั้นตอนของคำสั่งนั้นๆ เช่น คำสั่ง Select แสดงขั้นตอนดังภาพที่ 4.2 ดังต่อไปนี้



ภาพที่ 4.2 ตัวอย่างขั้นตอนการประมวลผลคำสั่ง Select [17]

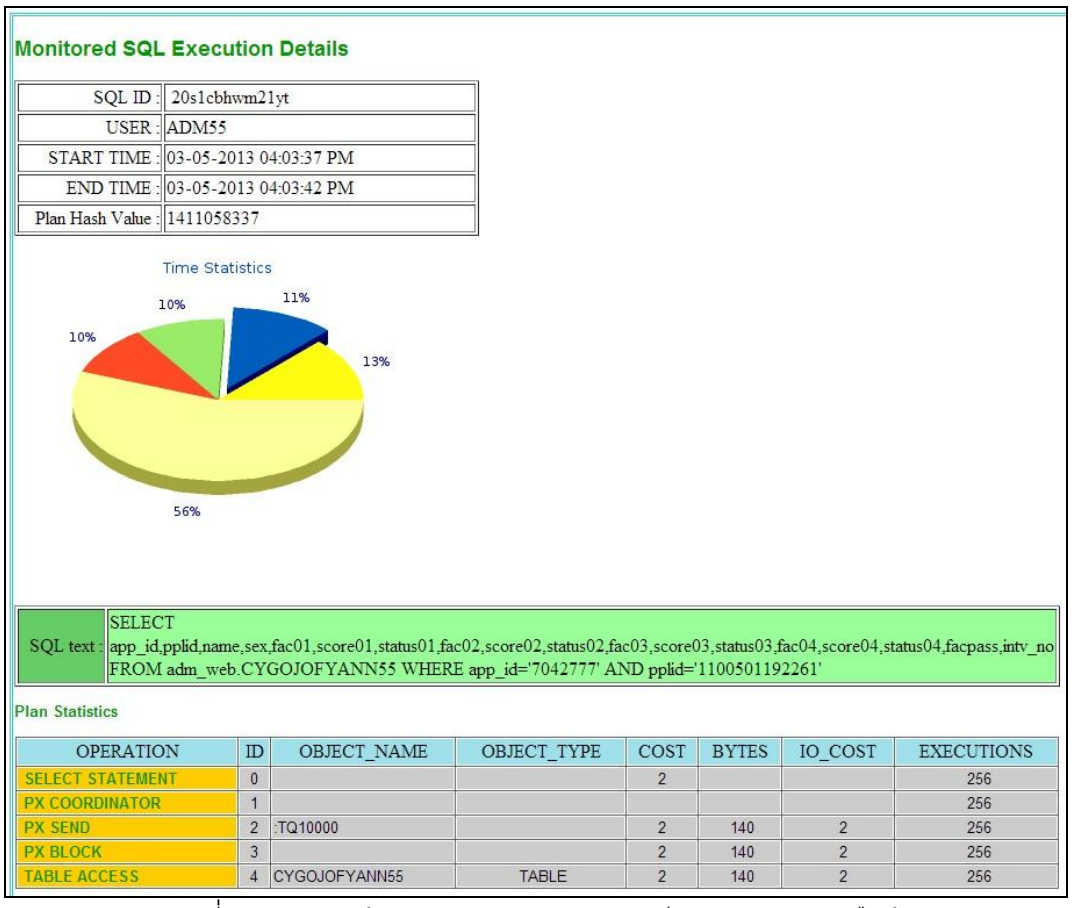
จากภาพที่ 4.2 แสดงขั้นตอนการประมวลผลคำสั่งเอสคิวแอล Select ประกอบด้วย 5 ขั้นตอน เริ่มจาก SELECT ตามด้วยพารามิเตอร์ที่ต้องการเรียกใช้แล้วตามด้วย FROM จากตาราง table_reference หรือ odbc_join_table และ WHERE ตามด้วยเงื่อนไขที่กำหนดให้แสดงผลของคำสั่งนั้นๆ ซึ่งสามารถจบคำสั่งและแสดงผลลัพธ์ได้ในขั้นตอนนี้ หรืออาจมีเพิ่มเติมอาร์กิวเมนต์ GROUP BY หรือ HAVING จะตรวจสอบวากยสัมพันธ์ตามรูปแบบที่ 4 และ 5

ส่วนแสดงผลของเครื่องมือจะแสดงในรูปแบบ เวลา, ซีพียู และ I/O ดังแสดงในภาพที่ 4.3

Status	SQL ID	User	Time	CPU	Concurrency	I/O	Buffer	Start	Ended	SQL Text
DONE	05s9358mm6vrr		7.412994 s (100%)	7.39 s (99.6 %)	0 s (0 %)	0.066929 s (0.90 %)	215232	15-02-2013 12:22:52 AM	15-02-2013 12:23:00 AM	begin dbms_feature_usage_internal.exec_db_usage_sampling(:bind1); end;
DONE	5zruc4v6y32f9	SYS	6.942914 s (100%)	6.57 s (94.6 %)	5.7E-5 s (0.00 %)	0.310881 s (4.47 %)	116958	14-02-2013 10:00:02 PM	14-02-2013 10:00:08 PM	DECLARE job BINARY_INTEGER := :job; next_date TIMESTAMP WITH TIME ZONE := :mydate; broken BOOLEAN := FALSE; job_name VARCHAR2(30) := :job_name; job_subname VARCHAR2(30) := :job_subname; job_owner VARCHAR2(30) := :job_owner; job_start TIMESTAMP WITH TIME ZONE := :job_start; job_scheduled_start TIMESTAMP WITH TIME ZONE := :job_scheduled_start; window_start TIMESTAMP WITH TIME ZONE := :window_start; window_end TIMESTAMP WITH TIME ZONE := :window_end; chain_id VARCHAR2(14) := :chainid; credential_owner varchar2(30) := :credown; credential_name varchar2(30) := :crednam; destination_owner varchar2(30) := :destown; destination_name varchar2(30) := :destnam; job_dest_id varchar2(14) := :jdestid; log_id number := :log_id; BEGIN DECLARE ename VARCHAR2(30); BEGIN ename := dbms_sqltune.execute_tuning_task('SYS_AUTO_SQL_TUNING_TASK'); END; :mydate := next_date; IF broken THEN :b := 1; ELSE :b := 0; END IF; END;
DONE	5zruc4v6y32f9	SYS	107.090166 s (100%)	106.79 s (99.7 %)	0.022437 s (0.02 %)	0.350166 s (0.32 %)	1093236	13-02-2013 10:00:02 PM	13-02-2013 10:01:48 PM	DECLARE job BINARY_INTEGER := :job; next_date TIMESTAMP WITH TIME ZONE := :mydate; broken BOOLEAN := FALSE; job_name VARCHAR2(30) := :job_name; job_subname VARCHAR2(30) := :job_subname; job_owner VARCHAR2(30) := :job_owner; job_start TIMESTAMP WITH TIME ZONE := :job_start; job_scheduled_start TIMESTAMP WITH TIME ZONE := :job_scheduled_start; window_start TIMESTAMP WITH TIME ZONE := :window_start; window_end TIMESTAMP WITH TIME ZONE := :window_end; chain_id VARCHAR2(14) := :chainid; credential_owner varchar2(30) := :credown; credential_name varchar2(30) := :crednam; destination_owner varchar2(30) := :destown; destination_name varchar2(30) := :destnam; job_dest_id varchar2(14) := :jdestid; log_id number := :log_id; BEGIN DECLARE ename VARCHAR2(30); BEGIN ename := dbms_sqltune.execute_tuning_task('SYS_AUTO_SQL_TUNING_TASK'); END; :mydate := next_date; IF broken THEN :b := 1; ELSE :b := 0; END IF; END;

ภาพที่ 4.3 แสดงหน้าจอการแสดงผลประสิทธิภาพของคำสั่งเอสคิวเอสในระบบ

จากภาพที่ 4.3 ส่วนแสดงผลประกอบด้วย เวลาที่ใช้ในการประมวลผลคำสั่งแต่ละครั้ง, ความเร็วของซีพียูที่ใช้, ร้อยละของ I/O, ขนาดของบัฟเฟอร์, เวลาเริ่มต้นและเวลาสิ้นสุดการประมวลผล และคำสั่งเอสคิวแอลที่ใช้ แต่ละรหัสคำสั่งเอสคิวแอลสามารถดูรายละเอียดของแผนการสืบค้นได้ ดังแสดงในภาพที่ 4.4 ดังนี้



Monitored SQL Execution Details

SQL ID :	20s1cbhwm21yt
USER :	ADM55
START TIME :	03-05-2013 04:03:37 PM
END TIME :	03-05-2013 04:03:42 PM
Plan Hash Value :	1411058337

Time Statistics

SQL text :	SELECT app_id,pplid,name,sex,fac01,score01,status01,fac02,score02,status02,fac03,score03,status03,fac04,score04,status04,facpass,intv_no FROM adm_web.CYGOJOFYANN55 WHERE app_id='7042777' AND pplid='1100501192261'

Plan Statistics

OPERATION	ID	OBJECT_NAME	OBJECT_TYPE	COST	BYTES	IO_COST	EXECUTIONS
SELECT STATEMENT	0			2			256
PX COORDINATOR	1						256
PX SEND	2	:TQ10000		2	140	2	256
PX BLOCK	3			2	140	2	256
TABLE ACCESS	4	CYGOJOFYANN55	TABLE	2	140	2	256

ภาพที่ 4.4 แสดงหน้าจอการแสดงผลรายละเอียดของแผนการสืบค้น

จากภาพที่ 4.4 แสดงหน้าจอการแสดงผลรายละเอียดของแผนการสืบค้นโดยระบุ ดังนี้

1) SQL ID คือ รหัสของคำสั่งเอสคิวแอล

2) USER คือ ผู้ใช้งานของคำสั่งเอสคิวแอลนี้

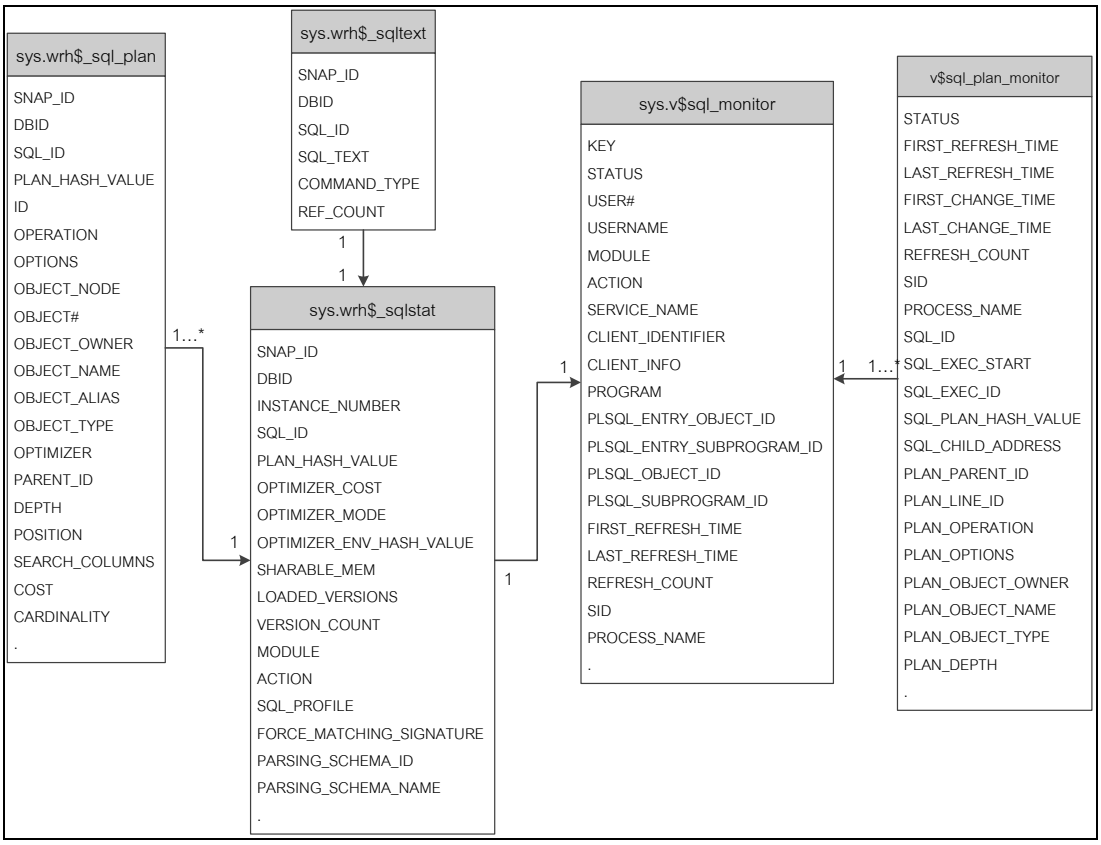
3) START TIME คือ เวลาเริ่มต้นในการประมวลผลคำสั่งเอสคิวแอล

4) END TIME คือ เวลาสิ้นสุดสำหรับการประมวลผลคำสั่งเอสคิวแอล

5) Plan hash value คือ รหัสตัวแปรของแผนการสืบค้น

4.3 โครงสร้างฐานข้อมูลของเครื่องมือ

โครงสร้างของฐานข้อมูลของเครื่องมือ อธิบายด้วยแผนภาพอีอาร์ ดังภาพที่ 4.5 โดยรายละเอียดเป็นดังนี้



ภาพที่ 4.5 โครงสร้างฐานข้อมูลของเครื่องมือ

จากภาพที่ 4.5 ประกอบด้วยตาราง ดังนี้

1. ตาราง wrh$_sql_plan เป็นตารางที่เก็บรายละเอียดของคำสั่งเอสคิวแอล
2. ตาราง wrh$_sqlstat เป็นตารางที่เก็บสถิติการใช้คำสั่งเอสคิวแอล
3. ตาราง wrh$_sqltext เป็นตารางที่เก็บคำสั่งเอสคิวแอล
4. ตาราง v$sql_monitor เป็นตารางที่เก็บการประมวลผลคำสั่งเอสคิวแอลทั้งหมด
5. ตาราง v$sql_plan_monitor เป็นตารางที่เก็บแผนการประมวลผลคำสั่งเอสคิวแอล

สำหรับพจนานุกรมข้อมูล ของแต่ละตารางสามารถดูได้ที่ ภาคผนวก ก

บทที่ 5

การทดสอบและประเมินผล

การทดสอบและประเมินผลนั้น เริ่มจากการกำหนดตัววัดประสิทธิภาพ ผลการปรับปรุงประสิทธิภาพของระบบกรณีศึกษา และผลการปรับปรุงคำสั่งเอสคิวแอล ซึ่งแสดงรายละเอียดดังต่อไปนี้

5.1 กำหนดตัววัดประสิทธิภาพ

ตัววัดประสิทธิภาพสำหรับการปรับปรุงประสิทธิภาพฐานข้อมูลของกรณีศึกษาในการวิเคราะห์ประสิทธิภาพ ได้แก่ การเลือกเกณฑ์หรือหลักการในการเปรียบเทียบประสิทธิภาพ ซึ่งตัววัดประสิทธิภาพของการประมวลผลของระบบ คือ เวลาในการประมวลผลของระบบ ซึ่งใช้เทคนิคการประเมินโดยการวัดจากระบบจริง และการวิเคราะห์ผลที่ได้จากการปรับปรุงประสิทธิภาพของระบบ

5.2 ผลการปรับปรุงประสิทธิภาพของระบบกรณีศึกษา

5.2.1 ผลการปรับปรุงระบบรับสมัครคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

ผลการปรับปรุงประสิทธิภาพของระบบ แสดงดังตารางที่ 5.1 ดังนี้

ตารางที่ 5.1 ผลการปรับปรุงประสิทธิภาพของระบบรับสมัครคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

การดำเนินการ	ผลลัพธ์	คำค้น	ผลการปรับปรุง		สรุปผล
			ก่อน (นาที)	หลัง (นาที)	
สร้างดรรชนีแบบ Reverse	✓	Q1	00:00:54	00:00:01	ลดลง 98.14%
	✓	Q2	00:00:54	00:00:01	ลดลง 98.14%
	✓	Q3	00:00:01	00:00:01	เท่าเดิม
	✓	Q4	00:00:54	00:00:01	ลดลง 98.14%
	✓	Q5	00:00:01	00:00:01	เท่าเดิม
	✓	Q6	00:00:01	00:00:01	เท่าเดิม

การดำเนินการ	ผลลัพธ์	คำค้น	ผลการปรับปรุง		สรุปผล
			ก่อน (นาที)	หลัง (นาที)	
ใช้ Parallel Hint	✓	Q7	00:00:37	00:00:21	ลดลง 43.24%
	✓	Q8	00:00:37	00:00:14	ลดลง 62.16%
	✓	Q9	00:00:01	00:00:01	เท่าเดิม

จากตารางที่ 5.1 จะเห็นว่าผลการปรับปรุงประสิทธิภาพของระบบรับสมัครคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษาดีขึ้นร้อยละ 44.42 จากการทดสอบด้วยการสร้างดรรชนีแบบ Reverse และการใช้ Parallel Hint ปรับปรุงคำสั่งเอสคิวแอลที่มีอยู่ในระบบจำนวน 9 คำค้น ดังแสดงคำค้นในตารางที่ 5.2 ดังนี้

ตารางที่ 5.2 แสดงคำสั่งเอสคิวแอลการปรับปรุงระบบรับสมัครคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

คำค้น	**ก่อนปรับปรุง**	**หลังปรับปรุง**
Q1	select shseq,pplid,name,sex,school, stuid10 from thesis_admhistory where pplid ='1100800980660';	select shseq,pplid,name,sex,school,stuid10 from thesis_admhistory where pplid ='1100800980660';
Q2	select shseq,oyear,oseat,gpseat253, gpseat353,gpseat154, gpseat155,gpseat255,pplid, name,sex,school,stuid10 from thesis_admhistory where shseq='552128080';	select shseq,oyear,oseat,gpseat253, gpseat353,gpseat154, gpseat155,gpseat255,pplid, name,sex,school,stuid10 from thesis_admhistory where shseq='552128080';
Q3	select schname ,schprov from school where schcode ='1224100000';	select schname ,schprov from school where schcode ='1224100000';
Q4	select oyear,oseat,gpseat253, gpseat353,gpseat154,gpseat155, gpseat255 from admhistory where shseq='552128080';	select oyear,oseat,gpseat253, gpseat353,gpseat154,gpseat155, gpseat255 from admhistory where shseq='552128080';

คำค้น	ก่อนปรับปรุง	หลังปรับปรุง
Q5	select admreg from register where shseq='542010829' and keydate='20120420' and keytime='12181695';	select admreg from register where shseq='542010829' and keydate='20120420' and keytime='12181695';
Q6	select name_fac, name_univ from factname where id_fac = '0002';	select name_fac, name_univ from factname where id_fac = '0002';
Q7	select admreg,shseq,pplid from register where admreg='7007964' and pplid='1361000223383';	select /*+ parallel(register 2) */ admreg,shseq,pplid from thesis_register where admreg='7007964' and pplid='1361000223383';
Q8	select admreg,shseq,pplid,name, sex,school,stuid10,fyear,fcen1,fcen2, fcen3,fcen4,billtotal,statuspay, paydate,paytime from register where admreg='7007964' and pplid='1361000223383';	select /*+ parallel(register 3) */ admreg, shseq,pplid,name,sex, school,stuid10,fyear,fcen1,fcen2,fcen3, fcen4,billtotal,statuspay,paydate,paytime from thesis_register where admreg='7007964' and pplid='1361000223383';
Q9	select name_fac, name_univ from factname where id_fac = '0002';	select /*+ parallel(factname 2) */ name_fac, name_univ from factname where id_fac = '0002';

5.2.2 ผลการปรับปรุงระบบรวมคะแนนและจัดลำดับการคัดเลือก

ผลการปรับปรุงประสิทธิภาพของระบบรวมคะแนนและจัดลำดับการคัดเลือก แสดงดังตารางที่ 5.3 ดังนี้

ตารางที่ 5.3 ผลการปรับปรุงประสิทธิภาพของระบบรวมคะแนนและจัดลำดับการคัดเลือก

การดำเนินการ	ผลลัพธ์	คำค้น	ผลการปรับปรุง		สรุปผล
			ก่อน (นาที)	หลัง (นาที)	
สร้างดรรชนีแบบ Reverse	✓	Q1	00:04:52	00:04:52	เท่าเดิม
	✓	Q2	00:04:52	00:04:52	เท่าเดิม
	✓	Q3	00:01:34	00:00:25	ลดลง 73.40%
	✓	Q4	00:01:35	00:00:28	ลดลง 70.52%
	✓	Q5	00:01:34	00:00:25	ลดลง 73.40%
ใช้ Parallel Hint	✓	Q6	00:00:24	00:00:01	ลดลง 95.83%
	✓	Q7	00:00:24	00:00:01	ลดลง 95.83%
	✓	Q8	00:00:24	00:00:24	เท่าเดิม

จากตารางที่ 5.3 จะเห็นว่าผลการปรับปรุงประสิทธิภาพของระบบรับสมัครคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษาดีขึ้นร้อยละ 51.12 (Q1, Q2 และ Q8 มีการสร้างดรรชนีแล้วทำให้การประมวลผลใช้เวลาเท่าเดิม) จากการทดสอบด้วยการสร้างดรรชนีแบบ Reverse และการใช้ Parallel Hint ปรับปรุงคำสั่งเอสคิวแอลที่มีอยู่ในระบบจำนวน 8 คำค้น ดังแสดงคำค้นในตารางที่ 5.4 ดังนี้

ตารางที่ 5.4 แสดงคำสั่งเอสคิวแอลการปรับปรุงระบบรวมคะแนนและจัดลำดับการคัดเลือก

คำค้น	**ก่อนปรับปรุง**	**หลังปรับปรุง**
Q1	update dbsubj set result_subjl='x' where dbsubj_rscore is null or dbsubj_rscore like '% %' or dbsubj_rscore like '%#%' or dbsubj_rscore like '%-%' or (dbsubj_rscore='-0000' and dbsubj_subj='29');	update dbsubj set result_subjl='x' where dbsubj_rscore is null or dbsubj_rscore like '% %' or dbsubj_rscore like '%#%' or dbsubj_rscore like '%-%' or (dbsubj_rscore='-0000' and dbsubj_subj='29');

คำค้น	ก่อนปรับปรุง	หลังปรับปรุง
Q2	update dbsubj set result_subjl='x' where result_subjl is null and to_number(dbsubj_sumg)< to_number(dbsubj_crt);	update dbsubj set result_subjl='x' where result_subjl is null and to_number(dbsubj_sumg)< to_number(dbsubj_crt);
Q3	update dbgrp set result_grp1='X' where to_number(dbgrp_sumg1)< to_number(dbgrp_crt1);	update dbgrp set result_grp1='X' where to_number(dbgrp_sumg1)< to_number(dbgrp_crt1);
Q4	update dbgrp set result_grp1='X' where to_number(dbgrp_Tscore)< to_number(dbgrp_TMMscore);	update dbgrp set result_grp1='X' where to_number(dbgrp_Tscore)< to_number(dbgrp_TMMscore);
Q5	update dbgrp set result_grpt='X' where dbgrp_TMMscore is not null and dbgrp_Tscore is not null and to_number(dbgrp_Tscore)< to_number(dbgrp_TMMscore);	update dbgrp set result_grpt='X' where dbgrp_TMMscore is not null and dbgrp_Tscore is not null and to_number(dbgrp_Tscore)< to_number(dbgrp_TMMscore);
Q6	update dbgrd set RESULT_GRP='X' where dbm_ty||dbm_sh_seq||dbm_appid|| dbgrd_ch||dbgrd_fac in (select dbm_ty||dbm_sh_seq||dbm_appid|| dbgrp_ch||dbgrp_fac from mam.adm56s_dbgrp_del);	update /*+ parallel(dbgrd 10) */ dbgrd set RESULT_GRP='X' where dbm_ty||dbm_sh_seq||dbm_appid|| dbgrd_ch||dbgrd_fac in (select dbm_ty||dbm_sh_seq||dbm_appid|| dbgrp_ch||dbgrp_fac from mam.adm56s_dbgrp_del);

คำค้น	ก่อนปรับปรุง	หลังปรับปรุง
Q7	update dbgrd set RESULT_GRP='X' where RESULT_GRP is null and to_number(dbgrd_sumg)< to_number(dbgrd_crt);	update /*+ parallel(dbgrd 10) */ dbgrd set RESULT_GRP='X' where RESULT_GRP is null and to_number(dbgrd_sumg)< to_number(dbgrd_crt);
Q8	select * fromdbgrd where result_grp is null and dbm_paid='1';	select /*+ parallel(dbgrd 10) */ * from dbgrd where result_grp is null and dbm_paid='1';

5.2.3 ผลการปรับปรุงระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

ผลการปรับปรุงประสิทธิภาพของระบบผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา แสดงดังตารางที่ 5.5 ดังนี้

ตารางที่ 5.5 ผลการปรับปรุงประสิทธิภาพของระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

การดำเนินการ	ผลลัพธ์	คำค้น	ผลการปรับปรุง		สรุปผล
			ก่อน (นาที)	หลัง (นาที)	
สร้างดรรชนีแบบ Reverse	✓	Q1	00:00:11	00:00:01	ลดลง 90.09%
	✓	Q2	00:00:01	00:00:01	เท่าเดิม
	✓	Q3	00:00:01	00:00:01	เท่าเดิม
ใช้ Parallel Hint	✓	Q4	00:00:11	00:00:06	ลดลง 45.45%

จากตารางที่ 5.5 จะเห็นว่าผลการปรับปรุงประสิทธิภาพของระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษาดีขึ้นร้อยละ 33.89 จากการทดสอบด้วยการสร้างดรรชนีแบบ Reverse และการใช้ Parallel Hint ปรับปรุงคำสั่งเอสคิวแอลที่มีอยู่ในระบบจำนวน 4 คำค้น ดังแสดงคำค้นในตารางที่ 5.6 ดังนี้

ตารางที่ 5.6 แสดงคำสั่งเอสคิวแอลการปรับปรุงระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

คำค้น	ก่อนปรับปรุง	หลังปรับปรุง
Q1	select app_id,pplid,name,sex,fac01, score01,status01,fac02,score02, status02,fac03,score03,status03, fac04,score04,status04,facpass, intv_no from thesis_result where app_id='7007972' and pplid='1340700421675';	select app_id,pplid,name,sex,fac01, score01,status01,fac02,score02, status02,fac03,score03,status03, fac04,score04,status04,facpass, intv_no from thesis_result where app_id='7007972' and pplid='1340700421675';
Q2	select schname ,schprov from school where schcode ='1224100000';	select schname ,schprov from school where schcode ='1224100000';
Q3	select * from factname;	select name_fac, name_univ from factname;
Q4	select app_id,pplid,name,sex,fac01, score01,status01,fac02,score02, status02,fac03,score03,status03, fac04,score04,status04,facpass, intv_no from thesis_result where app_id='7007972' and pplid='1340700421675';	select /*+ parallel(thesis_result 2) */ app_id,pplid,name,sex,fac01, score01,status01,fac02,score02, status02,fac03,score03,status03, fac04,score04,status04,facpass, intv_no from thesis_result where app_id='7007972' and pplid='1340700421675';

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

จากการศึกษาและทดลองของวิทยานิพนธ์นี้ สามารถสรุปผลการวิจัย และข้อเสนอแนะในการพัฒนาต่อ ดังรายละเอียดต่อไปนี้

6.1 สรุปผลการวิจัย

6.1.1 วิทยานิพนธ์นี้ได้นำเสนอวิธีการปรับปรุงประสิทธิภาพของระบบที่มีฐานข้อมูลขนาดใหญ่ การออกแบบเครื่องมือตรวจสอบสมรรถนะของฐานข้อมูล และเสนอแนะแนวทางการปรับปรุงฐานข้อมูลที่เหมาะสมของระบบที่ใช้ฐานข้อมูลของกรณีศึกษา ซึ่งได้ทดสอบกับระบบรับบุคคลเข้าศึกษาในสถาบันอุดมศึกษา ประกอบด้วยระบบย่อย 3 ระบบ คือ ระบบการรับสมัครบุคคลเข้าศึกษาในสถาบันอุดมศึกษา ระบบการรวมคะแนนและจัดลำดับการคัดเลือก และระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา ผลลัพธ์ของงานวิจัยนี้ได้เครื่องมือแสดงผลประสิทธิภาพของฐานข้อมูลออราเคิล ประสิทธิภาพที่ได้จากการปรับปรุงฐานข้อมูลที่เหมาะสมเปรียบเทียบกับประสิทธิภาพของฐานข้อมูลเดิมของระบบกรณีศึกษา คือ ระบบการรับสมัครบุคคลเข้าศึกษาในสถาบันอุดมศึกษาดีขึ้น 44.42% ระบบการรวมคะแนนและจัดลำดับการคัดเลือกดีขึ้น 51.12% และระบบประกาศผลการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษาดีขึ้น 33.89%

6.1.2 การเพิ่มประสิทธิภาพของการประมวลผลคำสั่งเอสคิวแอลขึ้นอยู่กับการใช้ดรรชนีด้วย ซึ่งดรรชนีมีการสร้างหลายแบบด้วยกัน สำหรับงานวิจัยนี้ใช้แบบ Bitmap และ Reverse จากผลการทดลองจะเห็นว่าการสร้างดรรชนีแบบ Bitmap และ Reverse จะช่วยให้ประสิทธิภาพของคำสั่งเอสคิวแอลเร็วขึ้น

6.1.3 การใช้ Parallel Hint ช่วยให้การสืบค้นข้อมูลในตารางที่มีขนาดใหญ่มีประสิทธิภาพในการสืบค้นเพิ่มขึ้น

6.2 ข้อเสนอแนะ

6.2.1 สำหรับการวิจัยนี้ได้ทดลองกับฐานข้อมูลออราเคิลเวอร์ชัน 11 จี เท่านั้น สามารถนำไปประยุกต์ใช้กับฐานข้อมูลอื่นได้ เช่น มายเอสคิวแอล (MySQL) เป็นต้น

6.2.2 สำหรับวิธีการจากวิทยานิพนธ์นี้สามารถนำไปประยุกต์ใช้กับระบบอื่นที่มีลักษณะการใช้ฐานข้อมูลเดียวกันได้ และเครื่องมือสามารถนำไปพัฒนาต่อให้การแสดงผลที่มีรายละเอียดยิ่งขึ้นได้

รายการอ้างอิง

[1] Ramez Elmasri and Shamkant B. Navathe. Fundamentals of Database Systems. Fifth Edition : Pearson Education, 2007.

[2] Bob Bryla and Kevin Loney. Oracle Database 11g DBA Handbook. Mc Graw-Hill, 2008.

[3] Kevin Jernigan and Adam Lee. Database Smart Flash Cache. Sun oracle, 2010.

[4] สำนักงานคณะกรรมการการอุดมศึกษา. ระเบียบการคัดเลือกบุคคลเข้าศึกษาในสถาบันอุดมศึกษา ประจำปีการศึกษา 2549., 2549.

[5] Donald K. Burleson. Oracle Tuning The Definitive Reference. Rampant TechPress, 2010.

[6] ลิสา สิมะสาธิตกุล. เครื่องมือสร้างสคีมาของฐานข้อมูลเชิงสัมพันธ์ จากแผนภาพอีอีอาร์, วิทยานิพนธ์ปริญญามหาบัณฑิต, สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย, 2555.

[7] ญาณี กาชัย. DBA Complete Guide Book ORACLE 11g. นนทบุรี : ไอดีซีฯ, 2551.

[8] Oracle Database standard edition one. Retrieved [Online].2012. Available from http://www.mindphp.com/คู่มือ/73-คืออะไร/2283-oracle-คืออะไร.html [2012,July 25]

[9] Berkovic, Z. Ivankovic, B. Markoski, D. Radosav, M. Ivkovic. Optimization of Bulk Operation Performances Within Oracle Database. IEEE 8th International Symposium on Intelligent Systems and Informatics 2010: 163-167.

[10] Chayaporn Kaensar. Analysis of the Query Plan for Assessing Optimizer Performance of SQL Select Statement. The Journal of KMUTNB 2012: 721-734.

[11] G. Rabinovitch and D. Wiese. Non-linear Optimization of Performance Functions for Autonomic Database Performance Tuning . IEEE International Conference on Autonomic and Autonomous Systems 2007.

[12] Dandan Li, Lu Han and Yi Ding. SQL Query Optimization Methods of Relational Database System. International Conference on Computer Engineering and Applications 2010: 557-560.

[13] G. Rabinovitch and D. Wiese. Knowledge Management in Autonomic Database Performance Tuning.

International Conference on Autonomic and Autonomous Systems 2009: 129-134.

[14] Tanakorn. การตรวจสอบ Performance โดยใช้ Enterprise Manager [Online]. 2012. Available from http://tanakornt.blogspot.com/2009/08/blog-post.html [2012,July 25]

[15] ORACLE DBA & IT.Retrieved [Online]. 2012. Available from http://oracle.jookku.com/2011/03/tip-choose-index/ [2012,July 25]

[16] P. Khaitan et al. Improved query plans for unnesting nested SQL queries. The 9th International Conference on Computer Science and its Applications 2009: 1-6.

[17] Oracle9i Lite SQL Reference [Online]. 2002. Available from http://docs.oracle.com/html/A95915_01/sqcmd.htm#i1009110 [2012,July 25]

[18] Using Parallel Execution [Online].2013. Available from http://docs.oracle.com/cd/B19306_01/server.102/b14223/usingpe.htm [2013,May 8]

[19] P. Belknap et at. Self-Tuning for SQL Performance in Oracle DB 11g. The 25th International Conference on Data Engineering 2009: 1321-1324.

[20] P. Belknap, B. Dageville, K. Dias and K. Yagoub. Self-Tuning for SQL Performance in Oracle Database 11g. IEEE International Conference on Data Engineering 2009: 1694-1700.

[21] Li Dandan et al. SQL Query Optimization Methods of Relational Database System. The 2nd International Conference on Computer Engineering and Applications 2010: 557-560.

[22] Jeong Seok Oh and Sang Ho Lee. Resource Selection for Autonomic Database Tuning. Proceedings of the 21st International Conference on Data Engineering 2005.

[23] Jason Price. Oracle Database 11g SQL. Mc Graw-Hill, 2008.

[24] Managing The Oracle Instance. Retrieved [Online].2013. Available from http://docs.oracle.com/cd/B28359_01/server.111/b28301/instance.htm [2013,May 6]

ภาคผนวก

ภาคผนวก ก

ภาคผนวก ก
พจนานุกรมข้อมูลของระบบกรณีศึกษาและเครื่องมือ

1. พจนานุกรมข้อมูลของระบบกรณีศึกษา ได้แก่

ตารางที่ ก-1 พจนานุกรมข้อมูลตาราง ONET_51

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
YEAR_COURSE	ปีที่จบ	CHAR(4)			
SEATNO	เลขที่นั่งสอบ	CHAR(8)			
PPLID	เลขประจำตัวประชาชน	CHAR(13)			
NAME	ชื่อ-นามสกุล	CHAR(40)			
SEX	เพศ	CHAR(1)			
SCHCODE	รหัสโรงเรียน	CHAR(10)			
SCHNAME	ชื่อโรงเรียน	CHAR(150)			
STUDENT_ID	รหัสนักเรียน	CHAR(13)			
PROVNAME	จังหวัด	CHAR(50)			
SCORE01	คะแนนวิชา 01	CHAR(10)			
SCORE02	คะแนนวิชา 02	CHAR(10)			
SCORE03	คะแนนวิชา 03	CHAR(10)			
SCORE04	คะแนนวิชา 04	CHAR(10)			
SCORE05	คะแนนวิชา 05	CHAR(10)			
SCORE06	คะแนนวิชา 06	CHAR(10)			
SCORE07	คะแนนวิชา 07	CHAR(10)			
SCORE08	คะแนนวิชา 08	CHAR(10)			

ตารางที่ ก-2 พจนานุกรมข้อมูลตาราง ONET_52

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
YEAR_COURSE	ปีที่จบ	CHAR(4)			
SEATNO	เลขที่นั่งสอบ	CHAR(8)			
PPLID	เลขประจำตัวประชาชน	CHAR(13)			
NAME	ชื่อ-นามสกุล	CHAR(40)			
SEX	เพศ	CHAR(1)			
SCHCODE	รหัสโรงเรียน	CHAR(10)			
SCHNAME	ชื่อโรงเรียน	CHAR(150)			
STUDENT_ID	รหัสนักเรียน	CHAR(13)			
PROVNAME	จังหวัด	CHAR(50)			
SCORE01	คะแนนวิชา 01	CHAR(10)			
SCORE02	คะแนนวิชา 02	CHAR(10)			
SCORE03	คะแนนวิชา 03	CHAR(10)			
SCORE04	คะแนนวิชา 04	CHAR(10)			
SCORE05	คะแนนวิชา 05	CHAR(10)			
SCORE06	คะแนนวิชา 06	CHAR(10)			
SCORE07	คะแนนวิชา 07	CHAR(10)			
SCORE08	คะแนนวิชา 08	CHAR(10)			

ตารางที่ ก-3 พจนานุกรมข้อมูลตาราง ONET_53

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
YEAR_COURSE	ปีที่จบ	CHAR(4)			
SEATNO	เลขที่นั่งสอบ	CHAR(8)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
PPLID	เลขประจำตัวประชาชน	CHAR(13)			
NAME	ชื่อ-นามสกุล	CHAR(40)			
SEX	เพศ	CHAR(1)			
SCHCODE	รหัสโรงเรียน	CHAR(10)			
SCHNAME	ชื่อโรงเรียน	CHAR(150)			
STUDENT_ID	รหัสนักเรียน	CHAR(13)			
PROVNAME	จังหวัด	CHAR(50)			
SCORE01	คะแนนวิชา 01	CHAR(10)			
SCORE02	คะแนนวิชา 02	CHAR(10)			
SCORE03	คะแนนวิชา 03	CHAR(10)			
SCORE04	คะแนนวิชา 04	CHAR(10)			
SCORE05	คะแนนวิชา 05	CHAR(10)			
SCORE06	คะแนนวิชา 06	CHAR(10)			
SCORE07	คะแนนวิชา 07	CHAR(10)			
SCORE08	คะแนนวิชา 08	CHAR(10)			

ตารางที่ ก-4 พจนานุกรมข้อมูลตาราง ONET_54

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
YEAR_COURSE	ปีที่จบ	CHAR(4)			
SEATNO	เลขที่นั่งสอบ	CHAR(8)			
PPLID	เลขประจำตัวประชาชน	CHAR(13)			
NAME	ชื่อ-นามสกุล	CHAR(40)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SEX	เพศ	CHAR(1)			
SCHCODE	รหัสโรงเรียน	CHAR(10)			
SCHNAME	ชื่อโรงเรียน	CHAR(150)			
STUDENT_ID	รหัสนักเรียน	CHAR(13)			
PROVNAME	จังหวัด	CHAR(50)			
SCORE01	คะแนนวิชา 01	CHAR(10)			
SCORE02	คะแนนวิชา 02	CHAR(10)			
SCORE03	คะแนนวิชา 03	CHAR(10)			
SCORE04	คะแนนวิชา 04	CHAR(10)			
SCORE05	คะแนนวิชา 05	CHAR(10)			
SCORE06	คะแนนวิชา 06	CHAR(10)			
SCORE07	คะแนนวิชา 07	CHAR(10)			
SCORE08	คะแนนวิชา 08	CHAR(10)			

ตารางที่ ก-5 พจนานุกรมข้อมูลตาราง GATPAT_353

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SEATNO	เลขที่นั่งสอบ	CHAR(8)			
PPLID	เลขประจำตัวประชาชน	CHAR(13)			
SCHCODE	รหัสโรงเรียน	CHAR(10)			
NAME	ชื่อ-นามสกุล	CHAR(50)			
GAT1	คะแนนวิชา GAT 1	CHAR(10)			
GAT2	คะแนนวิชา GAT 2	CHAR(10)			
GAT	คะแนนวิชา GAT	CHAR(10)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
PAT1	คะแนนวิชา PAT 1	CHAR(10)			
PAT2	คะแนนวิชา PAT 2	CHAR(10)			
PAT3	คะแนนวิชา PAT 3	CHAR(10)			
PAT4	คะแนนวิชา PAT 4	CHAR(10)			
PAT5	คะแนนวิชา PAT 5	CHAR(10)			
PAT6	คะแนนวิชา PAT 6	CHAR(10)			
PAT7_1	คะแนนวิชา PAT 7.1	CHAR(10)			
PAT7_2	คะแนนวิชา PAT 7.2	CHAR(10)			
PAT7_3	คะแนนวิชา PAT 7.3	CHAR(10)			
PAT7_4	คะแนนวิชา PAT 7.4	CHAR(10)			
PAT7_5	คะแนนวิชา PAT 7.5	CHAR(10)			
PAT7_6	คะแนนวิชา PAT 7.6	CHAR(10)			

ตารางที่ ก-6 พจนานุกรมข้อมูลตาราง GATPAT_154

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SEATNO	เลขที่นั่งสอบ	CHAR(8)			
PPLID	เลขประจำตัวประชาชน	CHAR(13)			
SCHCODE	รหัสโรงเรียน	CHAR(10)			
NAME	ชื่อ-นามสกุล	CHAR(50)			
GAT1	คะแนนวิชา GAT 1	CHAR(10)			
GAT2	คะแนนวิชา GAT 2	CHAR(10)			
GAT	คะแนนวิชา GAT	CHAR(10)			
PAT1	คะแนนวิชา PAT 1	CHAR(10)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
PAT2	คะแนนวิชา PAT 2	CHAR(10)			
PAT3	คะแนนวิชา PAT 3	CHAR(10)			
PAT4	คะแนนวิชา PAT 4	CHAR(10)			
PAT5	คะแนนวิชา PAT 5	CHAR(10)			
PAT6	คะแนนวิชา PAT 6	CHAR(10)			
PAT7_1	คะแนนวิชา PAT 7.1	CHAR(10)			
PAT7_2	คะแนนวิชา PAT 7.2	CHAR(10)			
PAT7_3	คะแนนวิชา PAT 7.3	CHAR(10)			
PAT7_4	คะแนนวิชา PAT 7.4	CHAR(10)			
PAT7_5	คะแนนวิชา PAT 7.5	CHAR(10)			
PAT7_6	คะแนนวิชา PAT 7.6	CHAR(10)			

ตารางที่ ก-7 พจนานุกรมข้อมูลตาราง GATPAT_155

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SEATNO	เลขที่นั่งสอบ	CHAR(8)			
PPLID	เลขประจำตัวประชาชน	CHAR(13)			
SCHCODE	รหัสโรงเรียน	CHAR(10)			
NAME	ชื่อ-นามสกุล	CHAR(50)			
GAT1	คะแนนวิชา GAT 1	CHAR(10)			
GAT2	คะแนนวิชา GAT 2	CHAR(10)			
GAT	คะแนนวิชา GAT	CHAR(10)			
PAT1	คะแนนวิชา PAT 1	CHAR(10)			
PAT2	คะแนนวิชา PAT 2	CHAR(10)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
PAT3	คะแนนวิชา PAT 3	CHAR(10)			
PAT4	คะแนนวิชา PAT 4	CHAR(10)			
PAT5	คะแนนวิชา PAT 5	CHAR(10)			
PAT6	คะแนนวิชา PAT 6	CHAR(10)			
PAT7_1	คะแนนวิชา PAT 7.1	CHAR(10)			
PAT7_2	คะแนนวิชา PAT 7.2	CHAR(10)			
PAT7_3	คะแนนวิชา PAT 7.3	CHAR(10)			
PAT7_4	คะแนนวิชา PAT 7.4	CHAR(10)			
PAT7_5	คะแนนวิชา PAT 7.5	CHAR(10)			
PAT7_6	คะแนนวิชา PAT 7.6	CHAR(10)			

ตารางที่ ก-8 พจนานุกรมข้อมูลตาราง GATPAT_255

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SEATNO	เลขที่นั่งสอบ	CHAR(8)			
PPLID	เลขประจำตัวประชาชน	CHAR(13)			
SCHCODE	รหัสโรงเรียน	CHAR(10)			
NAME	ชื่อ-นามสกุล	CHAR(50)			
GAT1	คะแนนวิชา GAT 1	CHAR(10)			
GAT2	คะแนนวิชา GAT 2	CHAR(10)			
GAT	คะแนนวิชา GAT	CHAR(10)			
PAT1	คะแนนวิชา PAT 1	CHAR(10)			
PAT2	คะแนนวิชา PAT 2	CHAR(10)			
PAT3	คะแนนวิชา PAT 3	CHAR(10)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
PAT4	คะแนนวิชา PAT 4	CHAR(10)			
PAT5	คะแนนวิชา PAT 5	CHAR(10)			
PAT6	คะแนนวิชา PAT 6	CHAR(10)			
PAT7_1	คะแนนวิชา PAT 7.1	CHAR(10)			
PAT7_2	คะแนนวิชา PAT 7.2	CHAR(10)			
PAT7_3	คะแนนวิชา PAT 7.3	CHAR(10)			
PAT7_4	คะแนนวิชา PAT 7.4	CHAR(10)			
PAT7_5	คะแนนวิชา PAT 7.5	CHAR(10)			
PAT7_6	คะแนนวิชา PAT 7.6	CHAR(10)			

พจนานุกรมข้อมูลของระบบรับสมัคร มีดังนี้

ตารางที่ ก-9 พจนานุกรมข้อมูลตาราง register

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
GENSEQ	เลขเรียงลำดับ	CHAR(10)		ไม่ว่าง	
ADMTIME	เวลาสมัคร	CHAR(1)			
ADMYEAR	ปีสมัคร	CHAR(4)			
ADMREG	เลขที่สมัคร	NUMBER(7)		ไม่ว่าง	
SP1	ที่ว่าง	CHAR(1)			
SHSEQ	เลขเรียงลำดับประวัติ	CHAR(9)			
OGROUP	กลุ่มโอเน็ต	CHAR(1)			
OYEAR	ปีโอเน็ต	CHAR(4)			
OSEAT	เลขที่นั่งสอบโอเน็ต	CHAR(8)			
GPTIME253	เวลา GAT/PAT	CHAR(1)			
GPYEAR253	ปีสอบ GAT/PAT	CHAR(4)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
GPSEAT253	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
GPTIME353	เวลา GAT/PAT	CHAR(1)			
GPYEAR353	ปีสอบ GAT/PAT	CHAR(4)			
GPSEAT353	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
GPTIME154	เวลา GAT/PAT	CHAR(1)			
GPYEAR154	ปีสอบ GAT/PAT	CHAR(4)			
GPSEAT154	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
GPTIME155	เวลา GAT/PAT	CHAR(1)			
GPYEAR155	ปีสอบ GAT/PAT	CHAR(4)			
GPSEAT155	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
GPTIME255	เวลา GAT/PAT	CHAR(1)			
GPYEAR255	ปีสอบ GAT/PAT	CHAR(4)			
GPSEAT255	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
SP2	ที่ว่าง	CHAR(1)			
PPLID	เลขประจำตัวประชาชน	CHAR(13)		ไม่ว่าง	
NAME	ชื่อ-นามสกุล	CHAR(40)			
SEX	เพศ	CHAR(1)			
SCHOOL	รหัสโรงเรียน	CHAR(10)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
STUID10	รหัสนักเรียน	CHAR(10)			
SP3	ที่ว่าง	CHAR(1)			
BIRTHDATE	วันเกิด	CHAR(8)			
FYEAR	ปี	CHAR(4)			
FSTATE	-	CHAR(1)			
SP4	ที่ว่าง	CHAR(1)			
GPAX	คะแนนเฉลี่ย	CHAR(4)			
GPA1_21	คะแนนเฉลี่ยวิชา 21	CHAR(4)			
GPA1_24	คะแนนเฉลี่ยวิชา 24	CHAR(4)			
GPA1_25	คะแนนเฉลี่ยวิชา 25	CHAR(4)			
GPA1_22	คะแนนเฉลี่ยวิชา 22	CHAR(4)			
GPA1_26	คะแนนเฉลี่ยวิชา 26	CHAR(4)			
GPA1_27	คะแนนเฉลี่ยวิชา 27	CHAR(4)			
GPA1_28	คะแนนเฉลี่ยวิชา 28	CHAR(4)			
GPA1_23	คะแนนเฉลี่ยวิชา 23	CHAR(4)			
SP5	ที่ว่าง	CHAR(1)			
FCEN1	คณะเลือกอันดับ 1	CHAR(4)		ไม่ว่าง	
FCEN1_STATUS	สถานะ 1	CHAR(1)			
FCEN2	คณะเลือกอันดับ 2	CHAR(4)			
FCEN2_STATUS	สถานะ 2	CHAR(1)			
FCEN3	คณะเลือกอันดับ 3	CHAR(4)			
FCEN3_STATUS	สถานะ 3	CHAR(1)			
FCEN4	คณะเลือกอันดับ 4	CHAR(4)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
FCEN4_STATUS	สถานะ 4	CHAR(1)			
BILLTOTAL	ค่าสมัคร	CHAR(4)			
ADDRESS	ที่อยู่	CHAR(100)			
TEL01	เบอร์โทร 1	CHAR(20)			
TEL02	เบอร์โทร 2	CHAR(20)			
KEYDATE	วันที่สมัคร	CHAR(8)		ไม่ว่าง	
KEYTIME	เวลาสมัคร	CHAR(8)		ไม่ว่าง	
PRNDATE	วันที่พิมพ์ใบสมัคร	CHAR(8)			
PRNTIME	เวลาพิมพ์ใบสมัคร	CHAR(8)			
FACROBAT	-	CHAR(15)			
STATUSPAY	สถานะชำระค่าสมัคร	CHAR(1)		ไม่ว่าง	
PAYDATE	วันชำระค่าสมัคร	CHAR(8)			
PAYTIME	เวลาชำระค่าสมัคร	CHAR(8)			
BANKCODE	รหัสธนาคาร	CHAR(3)			
BRANCHCODE	รหัสสาขาธนาคาร	CHAR(4)			
STATUS	สถานะ	CHAR(1)			

ตารางที่ ก-10 พจนานุกรมข้อมูลตาราง admhistory

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
GENSEQ	เลขเรียงลำดับ	CHAR(10)		ไม่ว่าง	
SP1	ที่ว่าง	CHAR(1)			
SHSEQ	เลขเรียงลำดับประวัติ	CHAR(9)		ไม่ว่าง	
OGROUP	กลุ่มโอเน็ต	NUMBER(1)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
OYEAR	ปีโอเน็ต	CHAR(4)			
OSEAT	เลขที่นั่งสอบโอเน็ต	CHAR(8)			
GPTIME253	เวลา GAT/PAT	CHAR(1)			
GPYEAR253	ปีสอบ GAT/PAT	CHAR(4)			
GPSEAT253	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
GPTIME353	เวลา GAT/PAT	CHAR(1)			
GPYEAR353	ปีสอบ GAT/PAT	CHAR(4)			
GPSEAT353	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
GPTIME154	เวลา GAT/PAT	CHAR(1)			
GPYEAR154	ปีสอบ GAT/PAT	CHAR(4)			
GPSEAT154	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
GPTIME155	เวลา GAT/PAT	CHAR(1)			
GPYEAR155	ปีสอบ GAT/PAT	CHAR(4)			
GPSEAT155	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
GPTIME255	เวลา GAT/PAT	CHAR(1)			
GPYEAR255	ปีสอบ GAT/PAT	CHAR(4)			
GPSEAT255	เลขที่นั่งสอบ GAT/PAT	CHAR(8)			
SP2	ที่ว่าง	CHAR(1)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
PPLID	เลขประจำตัวประชาชน	CHAR(13)		ไม่ว่าง	
NAME	ชื่อ-นามสกุล	CHAR(40)		ไม่ว่าง	
SEX	เพศ	CHAR(1)			
SCHOOL	รหัสโรงเรียน	CHAR(10)		ไม่ว่าง	
STUID10	รหัสนักเรียน	CHAR(10)			
SP3	ที่ว่าง	CHAR(1)			
FYEAR	ปีสมัคร	CHAR(4)			
FSTATE	-	CHAR(1)			
STATUS	สถานะ	CHAR(1)			

ตารางที่ ก-11 พจนานุกรมข้อมูลตาราง school

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SCHCODE	รหัสโรงเรียน	VARCHAR2(10)			
SCHNAME	ชื่อโรงเรียน	VARCHAR2(150)			
SCHPROV	จังหวัด	VARCHAR2(50)			
OLDSCHCODE	รหัสโรงเรียน 8 หลัก	CHAR(8)			
OLDSCHCODE10	รหัสโรงเรียนเดิม	CHAR(10)			
STATUS	สถานะ	CHAR(1)			

ตารางที่ ก-12 พจนานุกรมข้อมูลตาราง factname

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
ID_FAC	รหัสคณะ/สาขา	CHAR(4)			
NAME_FAC	ชื่อคณะ/สาขา	CHAR(100)			
ID_UNIV	รหัสมหาวิทยาลัย	CHAR(4)			
NAME_UNIV	ชื่อมหาวิทยาลัย	VARCHAR2(100)			

พจนานุกรมข้อมูลสำหรับระบบรวมคะแนน มีดังนี้

ตารางที่ ก-13 พจนานุกรมข้อมูลตาราง DBNM

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
DBM_TY	ลำดับเลือก	CHAR(1)			
DBM_SH_SEQ	เลขเรียงลำดับ	CHAR(9)			
DBM_APPID	เลขที่สมัคร	CHAR(7)			
DBM_PAID	สถานะใช้/ไม่ใช้	CHAR(1)			
DBM_CHOICE_0	ตัวเลือก	CHAR(1)			
DBPERSON_ID	เลขประจำตัวประชาชน	CHAR(13)			
DBPERSON_NAME	ชื่อ-นามสกุล	CHAR(40)			
DBGENDER	เพศ	CHAR(1)			
DBSCH_CODE	รหัสโรงเรียน	CHAR(10)			
DBSTU_ID	รหัสนักเรียน	CHAR(10)			
SP2	ที่ว่าง	CHAR(70)			
DBP_STAT	สถิติ	CHAR(3)			

ตารางที่ ก-14 พจนานุกรมข้อมูลตาราง DBSUBJ

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
DBM_PAID	สถานะใช้/ไม่ใช้	CHAR(1)			
DBSUBJ_CH	ลำดับที่เลือก	CHAR(1)			
DBSUBJ_FAC	รหัสคณะ	CHAR(4)			
DBSUBJ_GRP	เกณฑ์กลุ่มวิชา	CHAR(1)			
DBSUBJ_SUBJ	รหัสวิชา	CHAR(2)			
DBSUBJ_GAIN	-	CHAR(1)			
DBSUBJ_WG	ค่าน้ำหนัก	CHAR(5)			
DBSUBJ_FC	-	CHAR(5)			
DBSUBJ_RMAX	คะแนนสูงสุดดิบ	CHAR(9)			
DBSUBJ_MAX	คะแนนสูงสุด	CHAR(9)			
DBSUBJ_RSCORE	คะแนนดิบ	CHAR(9)			
DBSUBJ_SCORE	คะแนน	CHAR(9)			
DBSUBJ_MAXG	คะแนนสูงสุด	CHAR(9)			
DBSUBJ_SUMG	คะแนนรวม	CHAR(9)			
DBSUBJ_CRT	เกณฑ์ CRT	CHAR(9)			
DBSUBJ_RCRT	เกณฑ์คะแนนดิบ CRT	CHAR(5)			
FILLER	-	CHAR(60)			
RESULT_SUBJL	ผลลัพธ์	CHAR(3)			

ตารางที่ ก-15 พจนานุกรมข้อมูลตาราง DBGRP

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
DBM_TY	ลำดับเลือก	CHAR(1)			
DBM_SH_SEQ	เลขเรียงลำดับ	CHAR(9)			
DBM_APPID	เลขที่สมัคร	CHAR(7)			
DBM_PAID	สถานะใช้/ไม่ใช้	CHAR(1)			
DBGRP_CH	ลำดับที่เลือก	CHAR(1)			
DBGRP_FAC	รหัสคณะ	CHAR(4)			
DBGRP_GRP1	เกณฑ์กลุ่ม	CHAR(1)			
DBGRP_SUBJ1	เกณฑ์วิชา	CHAR(2)			
FILLER	-	CHAR(1)			
DBGRP_WG1	เกณฑ์ค่าน้ำหนัก	CHAR(5)			
DBGRP_FC1		CHAR(5)			
DBGRP_RMAX1	คะแนนสูงสุดดิบ	CHAR(9)			
DBGRP_MAX1	คะแนนสูงสุด	CHAR(9)			
DBGRP_RSCORE1	คะแนนดิบ	CHAR(9)			
DBGRP_SCORE1	คะแนน	CHAR(9)			
DBGRP_MAXG1	คะแนนสูงสุด	CHAR(9)			
DBGRP_SUMG1	คะแนนรวม	CHAR(9)			
DBGRP_CRT1	เกณฑ์ CRT	CHAR(9)			
DBGRP_RCRT1	เกณฑ์คะแนนดิบ CRT	CHAR(5)			
RESULT_GRP1	ผลลัพธ์	CHAR(3)			
DBGRP_GRP2	กลุ่ม	CHAR(1)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
DBGRP_SUBJ2	วิชา	CHAR(2)			
FILLER2	-	CHAR(1)			
DBGRP_WG2	ค่าน้ำหนัก	CHAR(5)			
DBGRP_FC2	-	CHAR(5)			
DBGRP_RMAX2	คะแนนสูงสุดดิบ	CHAR(9)			
DBGRP_MAX2	คะแนนสูงสุด	CHAR(9)			
DBGRP_RSCORE2	คะแนนดิบ	CHAR(9)			
DBGRP_SCORE2	คะแนน	CHAR(9)			
DBGRP_MAXG2	คะแนนสูงสุด	CHAR(9)			
DBGRP_SUMG2	คะแนนรวม	CHAR(9)			
DBGRP_CRT2	เกณฑ์ CRT	CHAR(9)			
DBGRP_RCRT2	เกณฑ์คะแนนดิบ CRT	CHAR(5)			
DBGRP_TMMSCORE	คะแนน	CHAR(9)			
DBGRP_TSCORE	คะแนน	CHAR(9)			
RESULT_GRPT	ผลลัพธ์	CHAR(3)			

ตารางที่ ก-16 พจนานุกรมข้อมูลตาราง DBGRD

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
DBM_TY	ลำดับเลือก	CHAR(1)			
DBM_SH_SEQ	เลขเรียงลำดับ	CHAR(9)			
DBM_APPID	เลขที่สมัคร	CHAR(7)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
DBM_PAID	สถานะใช้/ไม่ใช้	CHAR(1)			
DBGRD_CH	ลำดับที่เลือก	CHAR(1)			
DBGRD_FAC	รหัสคณะ	CHAR(4)			
DBGRD_GRP	เกณฑ์กลุ่ม	CHAR(1)			
DBGRD_SUBJ	เกณฑ์วิชา	CHAR(2)			
FILLER	-	CHAR(1)			
DBGRD_WG	เกณฑ์ค่าน้ำหนัก	CHAR(5)			
DBGRD_FC	-	CHAR(5)			
DBGRD_RMAX	คะแนนสูงสุดดิบ	CHAR(9)			
DBGRD_MAX	คะแนนสูงสุด	CHAR(9)			
DBGRD_RSCORE	คะแนนดิบ	CHAR(9)			
DBGRD_SCORE	คะแนน	CHAR(9)			
DBGRD_MAXG	คะแนนสูงสุด	CHAR(9)			
DBGRD_SUMG	คะแนนรวม	CHAR(9)			
DBGRD_CRT	เกณฑ์ CRT	CHAR(9)			
DBGRD_RCRT	เกณฑ์คะแนนดิบ CRT	CHAR(5)			
FILLER2	-	CHAR(46)			
RESULT_GRP	ผลลัพธ์	CHAR(3)			

ตารางที่ ก-17 พจนานุกรมข้อมูลตาราง DBSUBJ_DEL

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
DBM_TY	ลำดับเลือก	CHAR(1)			
DBM_SH_SEQ	เลขเรียงลำดับ	CHAR(9)			DBSUBJ
DBM_APPID	เลขที่สมัคร	CHAR(7)			
DBSUBJ_CH	ลำดับที่เลือก	CHAR(1)			
DBSUBJ_FAC	รหัสคณะ	CHAR(4)			

ตารางที่ ก-18 พจนานุกรมข้อมูลตาราง DBGRP_DEL

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
DBM_TY	ลำดับเลือก	CHAR(1)			
DBM_SH_SEQ	เลขเรียงลำดับ	CHAR(9)			DBGRP
DBM_APPID	เลขที่สมัคร	CHAR(7)			
DBGRP_CH	ลำดับที่เลือก	CHAR(1)			
DBGRP_FAC	รหัสคณะ	CHAR(4)			

ตารางที่ ก-19 พจนานุกรมข้อมูลตาราง SPT

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
APP_ID	เลขที่สมัคร	CHAR(7)			
FAC	รหัสคณะที่ผ่าน	CHAR(4)			
SCORE	คะแนน	VARCHAR2(10)			
CH	ลำดับที่เลือก	CHAR(1)			
FACPASS	รหัสคณะที่ผ่าน	CHAR(4)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
STATUS	สถานะ	CHAR(1)			

พจนานุกรมข้อมูลสำหรับระบบประกาศผล มีดังนี้

ตารางที่ ก-20 พจนานุกรมข้อมูลตาราง result

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
APP_ID	เลขที่สมัคร	CHAR(7)			
PPLID	เลขประจำตัวประชาชน	CHAR(13)			
NAME	ชื่อ-นามสกุล	CHAR(40)			
SEX	เพศ	CHAR(1)			
FAC_NO01	ลำดับคณะ1	CHAR(1)			
FAC01	รหัสคณะ1	CHAR(4)			
SCORE01	คะแนนที่ได้1	CHAR(9)			
MINSCORE01	คะแนนต่ำสุด1	CHAR(9)			
STATUS01	สถานะ1	CHAR(1)			
FAC_NO02	ลำดับคณะ2	CHAR(1)			
FAC02	รหัสคณะ2	CHAR(4)			
SCORE02	คะแนนที่ได้2	CHAR(9)			
MINSCORE02	คะแนนต่ำสุด2	CHAR(9)			
STATUS02	สถานะ2	CHAR(1)			
FAC_NO03	ลำดับคณะ3	CHAR(1)			
FAC03	รหัสคณะ3	CHAR(4)			
SCORE03	คะแนนที่ได้3	CHAR(9)			

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
MINSCORE03	คะแนนต่ำสุด3	CHAR(9)			
STATUS03	สถานะ3	CHAR(1)			
FAC_NO04	ลำดับคณะ4	CHAR(1)			
FAC04	รหัสคณะ4	CHAR(4)			
SCORE04	คะแนนที่ได้4	CHAR(9)			
MINSCORE04	คะแนนต่ำสุด4	CHAR(9)			
STATUS04	สถานะ4	CHAR(1)			
FACPASS	รหัสคณะที่ผ่าน	CHAR(4)			
INTV_NO	เลขสัมภาษณ์	CHAR(4)			

ตารางที่ ก-21 พจนานุกรมข้อมูลตาราง intview

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
ID_FAC	รหัสคณะ/สาขา	CHAR(4)			
NAME_FAC	ชื่อคณะ/สาขา	CHAR(100)			
ID_UNIV	รหัสมหาวิทยาลัย	CHAR(4)			
NAME_UNIV	ชื่อมหาวิทยาลัย	VARCHAR2(100)			
INTVIEW	สถานที่สัมภาษณ์	VARCHAR2(100)			

2. พจนานุกรมข้อมูลของเครื่องมือวัดประสิทธิภาพ ได้แก่

ตารางที่ ก-22 พจนานุกรมข้อมูลตาราง v$sql_monitor

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
KEY	NUMBER			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
STATUS	VARCHAR2(19)			
USER#	NUMBER			
USERNAME	VARCHAR2(30)			
MODULE	VARCHAR2(64)			
ACTION	VARCHAR2(64)			
SERVICE_NAME	VARCHAR2(64)			
CLIENT_IDENTIFIER	VARCHAR2(64)			
CLIENT_INFO	VARCHAR2(64)			
PROGRAM	VARCHAR2(48)			
PLSQL_ENTRY_OBJECT_ID	NUMBER			
PLSQL_ENTRY_SUBPROGRAM_ID	NUMBER			
PLSQL_OBJECT_ID	NUMBER			
PLSQL_SUBPROGRAM_ID	NUMBER			
FIRST_REFRESH_TIME	DATE			
LAST_REFRESH_TIME	DATE			
REFRESH_COUNT	NUMBER			
SID	NUMBER			
PROCESS_NAME	VARCHAR2(5)			
SQL_ID	VARCHAR2(13)			
SQL_TEXT	VARCHAR2(2000)			
IS_FULL_SQLTEXT	VARCHAR2(1)			
SQL_EXEC_START	DATE			
SQL_EXEC_ID	NUMBER			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SQL_PLAN_HASH_VALUE	NUMBER			
EXACT_MATCHING_SIGNATURE	NUMBER			
FORCE_MATCHING_SIGNATURE	NUMBER			
SQL_CHILD_ADDRESS	RAW(8)			
SESSION_SERIAL#	NUMBER			
PX_IS_CROSS_INSTANCE	VARCHAR2(1)			
PX_MAXDOP	NUMBER			
PX_MAXDOP_INSTANCES	NUMBER			
PX_SERVERS_REQUESTED	NUMBER			
PX_SERVERS_ALLOCATED	NUMBER			
PX_SERVER#	NUMBER			
PX_SERVER_GROUP	NUMBER			
PX_SERVER_SET	NUMBER			
PX_QCINST_ID	NUMBER			
PX_QCSID	NUMBER			
ERROR_NUMBER	VARCHAR2(40)			
ERROR_FACILITY	VARCHAR2(4)			
ERROR_MESSAGE	VARCHAR2(256)			
BINDS_XML	CLOB			
OTHER_XML	CLOB			
ELAPSED_TIME	NUMBER			
QUEUING_TIME	NUMBER			
CPU_TIME	NUMBER			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
FETCHES	NUMBER			
BUFFER_GETS	NUMBER			
DISK_READS	NUMBER			
DIRECT_WRITES	NUMBER			
IO_INTERCONNECT_BYTES	NUMBER			
PHYSICAL_READ_REQUESTS	NUMBER			
PHYSICAL_READ_BYTES	NUMBER			
PHYSICAL_WRITE_REQUESTS	NUMBER			
PHYSICAL_WRITE_BYTES	NUMBER			
APPLICATION_WAIT_TIME	NUMBER			
CONCURRENCY_WAIT_TIME	NUMBER			
CLUSTER_WAIT_TIME	NUMBER			
USER_IO_WAIT_TIME	NUMBER			
PLSQL_EXEC_TIME	NUMBER			
JAVA_EXEC_TIME	NUMBER			

ตารางที่ ก-23 พจนานุกรมข้อมูลตาราง v$sql_plan_monitor

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
STATUS	VARCHAR2(19)			
FIRST_REFRESH_TIME	DATE			
LAST_REFRESH_TIME	DATE			
FIRST_CHANGE_TIME	DATE			
LAST_CHANGE_TIME	DATE			
REFRESH_COUNT	NUMBER			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SID	NUMBER			
PROCESS_NAME	VARCHAR2(5)			
SQL_ID	VARCHAR2(13)			
SQL_EXEC_START	DATE			
SQL_EXEC_ID	NUMBER			
SQL_PLAN_HASH_VALUE	NUMBER			
SQL_CHILD_ADDRESS	RAW(8)			
PLAN_PARENT_ID	NUMBER			
PLAN_LINE_ID	NUMBER			
PLAN_OPERATION	VARCHAR2(30)			
PLAN_OPTIONS	VARCHAR2(30)			
PLAN_OBJECT_OWNER	VARCHAR2(30)			
PLAN_OBJECT_NAME	VARCHAR2(30)			
PLAN_OBJECT_TYPE	VARCHAR2(20)			
PLAN_DEPTH	NUMBER			
PLAN_POSITION	NUMBER			
PLAN_COST	NUMBER			
PLAN_CARDINALITY	NUMBER			
PLAN_BYTES	NUMBER			
PLAN_TIME	NUMBER			
PLAN_PARTITION_START	VARCHAR2(64)			
PLAN_PARTITION_STOP	VARCHAR2(64)			
PLAN_CPU_COST	NUMBER			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
PLAN_IO_COST	NUMBER			
PLAN_TEMP_SPACE	NUMBER			
STARTS	NUMBER			
OUTPUT_ROWS	NUMBER			
IO_INTERCONNECT_BYTES	NUMBER			
PHYSICAL_READ_REQUESTS	NUMBER			
PHYSICAL_READ_BYTES	NUMBER			
PHYSICAL_WRITE_REQUESTS	NUMBER			
PHYSICAL_WRITE_BYTES	NUMBER			
WORKAREA_MEM	NUMBER			
WORKAREA_MAX_MEM	NUMBER			
WORKAREA_TEMPSEG	NUMBER			
WORKAREA_MAX_TEMPSEG	NUMBER			

ตารางที่ ก-24 พจนานุกรมข้อมูลตาราง wrh$_sqltext

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SNAP_ID	NUMBER			
DBID	NUMBER			
SQL_ID	VARCHAR2(13)			
SQL_TEXT	CLOB			
COMMAND_TYPE	NUMBER			
REF_COUNT	NUMBER			

ตารางที่ ก-25 พจนานุกรมข้อมูลตาราง wrh$_sql_plan

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SNAP_ID	NUMBER			
DBID	NUMBER			
SQL_ID	VARCHAR2(13)			
PLAN_HASH_VALUE	NUMBER			
ID	NUMBER			
OPERATION	VARCHAR2(30)			
OPTIONS	VARCHAR2(30)			
OBJECT_NODE	VARCHAR2(128)			
OBJECT#	NUMBER			
OBJECT_OWNER	VARCHAR2(30)			
OBJECT_NAME	VARCHAR2(31)			
OBJECT_ALIAS	VARCHAR2(65)			
OBJECT_TYPE	VARCHAR2(20)			
OPTIMIZER	VARCHAR2(20)			
PARENT_ID	NUMBER			
DEPTH	NUMBER			
POSITION	NUMBER			
SEARCH_COLUMNS	NUMBER			
COST	NUMBER			
CARDINALITY	NUMBER			
BYTES	NUMBER			
OTHER_TAG	VARCHAR2(35)			
PARTITION_START	VARCHAR2(64)			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
PARTITION_STOP	VARCHAR2(64)			
PARTITION_ID	NUMBER			
OTHER	VARCHAR2(4000)			
DISTRIBUTION	VARCHAR2(20)			
CPU_COST	NUMBER			
IO_COST	NUMBER			
TEMP_SPACE	NUMBER			
ACCESS_PREDICATES	VARCHAR2(4000)			
FILTER_PREDICATES	VARCHAR2(4000)			
PROJECTION	VARCHAR2(4000)			
TIME	NUMBER			
QBLOCK_NAME	VARCHAR2(31)			
REMARKS	VARCHAR2(4000)			
TIMESTAMP	DATE			
OTHER_XML	CLOB			

ตารางที่ ก-26 พจนานุกรมข้อมูลตาราง wrh$_sqlstat

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
SNAP_ID	NUMBER			
DBID	NUMBER			
INSTANCE_NUMBER	NUMBER			
SQL_ID	VARCHAR2(13)			
PLAN_HASH_VALUE	NUMBER			
OPTIMIZER_COST	NUMBER			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
OPTIMIZER_MODE	VARCHAR2(10)			
OPTIMIZER_ENV_HASH_VALUE	NUMBER			
SHARABLE_MEM	NUMBER			
LOADED_VERSIONS	NUMBER			
VERSION_COUNT	NUMBER			
MODULE	VARCHAR2(64)			
ACTION	VARCHAR2(64)			
SQL_PROFILE	VARCHAR2(64)			
FORCE_MATCHING_SIGNATURE	NUMBER			
PARSING_SCHEMA_ID	NUMBER			
PARSING_SCHEMA_NAME	VARCHAR2(30)			
FETCHES_TOTAL	NUMBER			
FETCHES_DELTA	NUMBER			
END_OF_FETCH_COUNT_TOTAL	NUMBER			
END_OF_FETCH_COUNT_DELTA	NUMBER			
SORTS_TOTAL	NUMBER			
SORTS_DELTA	NUMBER			
EXECUTIONS_TOTAL	NUMBER			
EXECUTIONS_DELTA	NUMBER			
PX_SERVERS_EXECS_TOTAL	NUMBER			
PX_SERVERS_EXECS_DELTA	NUMBER			
LOADS_TOTAL	NUMBER			
LOADS_DELTA	NUMBER			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
INVALIDATIONS_TOTAL	NUMBER			
INVALIDATIONS_DELTA	NUMBER			
PARSE_CALLS_TOTAL	NUMBER			
PARSE_CALLS_DELTA	NUMBER			
DISK_READS_TOTAL	NUMBER			
DISK_READS_DELTA	NUMBER			
BUFFER_GETS_TOTAL	NUMBER			
BUFFER_GETS_DELTA	NUMBER			
ROWS_PROCESSED_TOTAL	NUMBER			
ROWS_PROCESSED_DELTA	NUMBER			
CPU_TIME_TOTAL	NUMBER			
CPU_TIME_DELTA	NUMBER			
ELAPSED_TIME_TOTAL	NUMBER			
ELAPSED_TIME_DELTA	NUMBER			
IOWAIT_TOTAL	NUMBER			
IOWAIT_DELTA	NUMBER			
CLWAIT_TOTAL	NUMBER			
CLWAIT_DELTA	NUMBER			
APWAIT_TOTAL	NUMBER			
APWAIT_DELTA	NUMBER			
CCWAIT_TOTAL	NUMBER			
CCWAIT_DELTA	NUMBER			
DIRECT_WRITES_TOTAL	NUMBER			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
DIRECT_WRITES_DELTA	NUMBER			
PLSEXEC_TIME_TOTAL	NUMBER			
PLSEXEC_TIME_DELTA	NUMBER			
JAVEXEC_TIME_TOTAL	NUMBER			
JAVEXEC_TIME_DELTA	NUMBER			
BIND_DATA	RAW(2000)			
FLAG	NUMBER			
PARSING_USER_ID	NUMBER			
IO_OFFLOAD_ELIG_BYTES_TOTAL	NUMBER			
IO_OFFLOAD_ELIG_BYTES_DELTA	NUMBER			
IO_INTERCONNECT_BYTES_TOTAL	NUMBER			
IO_INTERCONNECT_BYTES_DELTA	NUMBER			
PHYSICAL_READ_REQUESTS_TOTAL	NUMBER			
PHYSICAL_READ_REQUESTS_DELTA	NUMBER			
PHYSICAL_READ_BYTES_TOTAL	NUMBER			
PHYSICAL_READ_BYTES_DELTA	NUMBER			
PHYSICAL_WRITE_REQUESTS_TOTAL	NUMBER			
PHYSICAL_WRITE_REQUESTS_DELTA	NUMBER			
PHYSICAL_WRITE_BYTES_TOTAL	NUMBER			
PHYSICAL_WRITE_BYTES_DELTA	NUMBER			
OPTIMIZED_PHYSICAL_READS_TOTAL	NUMBER			
OPTIMIZED_PHYSICAL_READS_DELTA	NUMBER			
CELL_UNCOMPRESSED_BYTES_TOTAL	NUMBER			

ชื่อ	ชนิดข้อมูล	คีย์	ค่าว่าง	ตารางที่อ้างอิง
CELL_UNCOMPRESSED_BYTES_DELTA	NUMBER			
IO_OFFLOAD_RETURN_BYTES_TOTAL	NUMBER			
IO_OFFLOAD_RETURN_BYTES_DELTA	NUMBER			

ภาคผนวก ข

ภาคผนวก ข
เอกสารเอสคิวแอล

1. ระบบการรวมคะแนนและจัดลำดับผู้ผ่านเกณฑ์การคัดเลือก

เอกสารเอสคิวแอลของระบบการรวมคะแนนและจัดลำดับผู้ผ่านเกณฑ์การคัดเลือก เป็นดังนี้

```
DROP TABLE ADM55S_DBNM PURGE;
DROP TABLE ADM55S_DBSUBJ PURGE;
DROP TABLE ADM55S_DBGRP PURGE;
DROP TABLE ADM55S_DBGRD PURGE;
DROP TABLE ADM55S_DBSUBJ_DEL PURGE;
DROP TABLE ADM55S_DBGRP_DEL PURGE;
DROP TABLE ADM55S_SPT PURGE;

CREATE TABLE ADM55S_DBNM (
DBM_TY       CHAR(1),
DBM_SH_SEQ    CHAR(9),
DBM_APPID     CHAR(7),
DBM_PAID      CHAR(1),
DBM_CHOICE_0  CHAR(1),
DBPERSON_ID   CHAR(13),
DBPERSON_NAME CHAR(40),
DBGENDER      CHAR(1),
DBSCH_CODE    CHAR(10),
DBSTU_ID      CHAR(10),
SP2          CHAR(70),
DBP_STAT      CHAR(3));

CREATE TABLE ADM55S_DBSUBJ (
DBM_TY       CHAR(1),
DBM_SH_SEQ    CHAR(9),
```

```
DBM_APPID     CHAR(7),
DBM_PAID      CHAR(1),
DBSUBJ_CH          CHAR(1),
DBSUBJ_FAC         CHAR(4),
DBSUBJ_GRP         CHAR(1),
DBSUBJ_SUBJ        CHAR(2),
DBSUBJ_GAIN        CHAR(1),
DBSUBJ_WG          CHAR(5),
DBSUBJ_FC          CHAR(5),
DBSUBJ_RMAX        CHAR(9),
DBSUBJ_MAX         CHAR(9),
DBSUBJ_RSCORE      CHAR(9),
DBSUBJ_SCORE       CHAR(9),
DBSUBJ_MAXG        CHAR(9),
DBSUBJ_SUMG        CHAR(9),
DBSUBJ_CRT         CHAR(9),
DBSUBJ_RCRT        CHAR(5),
FILLER             CHAR(60),
RESULT_SUBJL       CHAR(3));

CREATE TABLE ADM55S_DBGRP (
DBM_TY        CHAR(1),
DBM_SH_SEQ    CHAR(9),
DBM_APPID     CHAR(7),
DBM_PAID      CHAR(1),
DBGRP_CH          CHAR(1),
DBGRP_FAC         CHAR(4),
DBGRP_GRP1        CHAR(1),
DBGRP_SUBJ1       CHAR(2),
FILLER            CHAR(1),
DBGRP_WG1         CHAR(5),
```

```
DBGRP_FC1        CHAR(5),
DBGRP_RMAX1      CHAR(9),
DBGRP_MAX1       CHAR(9),
DBGRP_RSCORE1    CHAR(9),
DBGRP_SCORE1     CHAR(9),
DBGRP_MAXG1      CHAR(9),
DBGRP_SUMG1      CHAR(9),
DBGRP_CRT1       CHAR(9),
DBGRP_RCRT1      CHAR(5),
RESULT_GRP1      CHAR(3),
DBGRP_GRP2       CHAR(1),
DBGRP_SUBJ2      CHAR(2),
FILLER2          CHAR(1),
DBGRP_WG2        CHAR(5),
DBGRP_FC2        CHAR(5),
DBGRP_RMAX2      CHAR(9),
DBGRP_MAX2       CHAR(9),
DBGRP_RSCORE2    CHAR(9),
DBGRP_SCORE2     CHAR(9),
DBGRP_MAXG2      CHAR(9),
DBGRP_SUMG2      CHAR(9),
DBGRP_CRT2       CHAR(9),
DBGRP_RCRT2      CHAR(5),
DBGRP_TMMSCORE   CHAR(9),
DBGRP_TSCORE     CHAR(9),
RESULT_GRPT      CHAR(3));

CREATE TABLE ADM55S_DBGRD (
DBM_TY       CHAR(1),
DBM_SH_SEQ   CHAR(9),
DBM_APPID    CHAR(7),
```

```
DBM_PAID      CHAR(1),
DBGRD_CH          CHAR(1),
DBGRD_FAC         CHAR(4),
DBGRD_GRP         CHAR(1),
DBGRD_SUBJ        CHAR(2),
FILLER            CHAR(1),
DBGRD_WG          CHAR(5),
DBGRD_FC          CHAR(5),
DBGRD_RMAX        CHAR(9),
DBGRD_MAX         CHAR(9),
DBGRD_RSCORE      CHAR(9),
DBGRD_SCORE       CHAR(9),
DBGRD_MAXG        CHAR(9),
DBGRD_SUMG        CHAR(9),
DBGRD_CRT         CHAR(9),
DBGRD_RCRT        CHAR(5),
FILLER2           CHAR(46),
RESULT_GRP        CHAR(3));

HO SQLLDR MAM/XXXXXX DBGRP.CTL
HO SQLLDR MAM/XXXXXX DBGRD.CTL
HO SQLLDR MAM/XXXXXX DBSUBJ.CTL
HO SQLLDR MAM/XXXXXX DBNM.CTL

UPDATE ADM55S_DBSUBJ SET DBSUBJ_RSCORE='000000000'
WHERE
DBSUBJ_SUBJ='29' AND
(DBSUBJ_RSCORE LIKE '% %' OR DBSUBJ_RSCORE LIKE '%#%');

UPDATE  ADM55S_DBSUBJ
SET RESULT_SUBJL='X'
```

```
WHERE
DBSUBJ_RSCORE IS NULL OR
DBSUBJ_RSCORE LIKE '% %' OR
DBSUBJ_RSCORE LIKE '%#%' OR
DBSUBJ_RSCORE LIKE '%-%';
COMMIT;

UPDATE  ADM55S_DBSUBJ
SET RESULT_SUBJL='X'
WHERE
RESULT_SUBJL IS NULL AND
TO_NUMBER(DBSUBJ_SUMG)<TO_NUMBER(DBSUBJ_CRT);
COMMIT;

CREATE TABLE ADM55S_DBSUBJ_DEL AS
SELECT DISTINCT
DBM_TY,
DBM_SH_SEQ,
DBM_APPID,
DBSUBJ_CH,
DBSUBJ_FAC
FROM ADM55S_DBSUBJ
WHERE
RESULT_SUBJL='X'
ORDER BY DBM_APPID,DBSUBJ_CH;

UPDATE ADM55S_DBGRD
SET RESULT_GRP='X'
WHERE
DBM_TY||DBM_SH_SEQ||DBM_APPID||DBGRD_CH||DBGRD_FAC IN
(SELECT DBM_TY||DBM_SH_SEQ||DBM_APPID||DBSUBJ_CH||DBSUBJ_FAC FROM
```

```
ADM55S_DBSUBJ_DEL);

UPDATE  ADM55S_DBGRP
SET RESULT_GRP1='X'
WHERE
TO_NUMBER(DBGRP_SUMG1)<TO_NUMBER(DBGRP_CRT1);
COMMIT;

UPDATE  ADM55S_DBGRP
SET RESULT_GRP1='X'
WHERE
TO_NUMBER(DBGRP_TSCORE)<TO_NUMBER(DBGRP_TMMSCORE);
COMMIT;

UPDATE  ADM55S_DBGRP
SET RESULT_GRPT='X'
WHERE
DBGRP_TMMSCORE IS NOT NULL AND DBGRP_TSCORE IS NOT NULL AND
TO_NUMBER(DBGRP_TSCORE)<TO_NUMBER(DBGRP_TMMSCORE);
COMMIT;

CREATE TABLE ADM55S_DBGRP_DEL AS
SELECT DISTINCT
DBM_TY,
DBM_SH_SEQ,
DBM_APPID,
DBGRP_CH,
DBGRP_FAC
FROM ADM55S_DBGRP
WHERE
RESULT_GRP1='X' OR
```

```
RESULT_GRPT='X'
ORDER BY DBM_TY,DBM_SH_SEQ,DBM_APPID,DBGRP_CH,DBGRP_FAC;
COMMIT;

UPDATE ADM55S_DBGRD
SET RESULT_GRP='X'
WHERE
DBM_TY||DBM_SH_SEQ||DBM_APPID||DBGRD_CH||DBGRD_FAC IN
(SELECT DBM_TY||DBM_SH_SEQ||DBM_APPID||DBGRP_CH||DBGRP_FAC FROM
ADM55S_DBGRP_DEL);
COMMIT;
UPDATE ADM55S_DBGRD
SET RESULT_GRP='X'
WHERE
RESULT_GRP IS NULL AND
TO_NUMBER(DBGRD_SUMG)<TO_NUMBER(DBGRD_CRT);
COMMIT;

SELECT COUNT(*) FROM ADM55S_DBGRD
WHERE
RESULT_GRP IS NULL AND DBM_PAID='1';

CREATE TABLE ADM55S_SPT AS
SELECT
DBM_APPID APP_ID,DBGRD_FAC FAC,
SUBSTR(DBGRD_SUMG,1,5)||'.'||SUBSTR(DBGRD_SUMG,6,4) SCORE,
DBGRD_CH CH,'0000' FACPASS FROM ADM55S_DBGRD
WHERE
RESULT_GRP IS NULL  AND DBM_PAID='1'
ORDER BY  DBGRD_CH,DBGRD_SCORE DESC;
ALTER TABLE ADM55S_SPT ADD STATUS CHAR(1);
```

2. ระบบรับสมัครบุคคลเข้าศึกษาในสถาบันอุดมศึกษา

เอกสารเอสคิวแอลของระบบรับสมัคร เป็นดังนี้

Page1:

```
SELECT SHSEQ,PPLID,NAME,SEX,SCHOOL,STUID10 FROM ADMHISTORY WHERE PPLID
='$PPLID_INPUT';
```

Page2:

```
SELECT
SHSEQ,OYEAR,OSEAT,GPSEAT253,GPSEAT353,GPSEAT154,GPSEAT155,GPSEAT255,PPLID,NAME,SE
X,SCHOOL,STUID10 FROM ADMHISTORY  WHERE SHSEQ='$SHSEQ_CHECK';
SELECT SCHNAME ,SCHPROV FROM SCHOOL WHERE SCHCODE ='$SCHOOL_INPUT';
```

PAGE3:

```
SELECT  OYEAR,OSEAT,GPSEAT253,GPSEAT353,GPSEAT154,GPSEAT155,GPSEAT255  FROM
ADMHISTORY  WHERE SHSEQ='$SHSEQ_CHECK';
SELECT SCHNAME ,SCHPROV FROM SCHOOL WHERE SCHCODE ='$SCHOOL_CHECK';
```

PAGE4:

```
SELECT  GENSEQ ,OYEAR,OSEAT,GPSEAT253,GPSEAT353,GPSEAT154,GPSEAT155,GPSEAT255
FROM ADMHISTORY  WHERE SHSEQ='$SHSEQ_CHECK';


INSERT INTO REGISTER (GENSEQ,ADMTIME,ADMYEAR,SHSEQ,OGROUP,OYEAR,OSEAT,
GPTIME253,GPYEAR253,GPSEAT253,
GPTIME353,GPYEAR353,GPSEAT353,
GPTIME154,GPYEAR154,GPSEAT154,
GPTIME155,GPYEAR155,GPSEAT155,
GPTIME255,GPYEAR255,GPSEAT255,
PPLID,NAME,SEX,SCHOOL,STUID10,BIRTHDATE,FYEAR,FSTATE,GPAX,FCEN1,FCEN2,FCEN3,FCEN4,
ADDRESS,TEL01,TEL02,KEYDATE,KEYTIME,BILLTOTAL,STATUSPAY,STATUS) VALUES ('';
```

```
$SQL = $SQL.$GENSEQ ."','1','2555','".$SHSEQ_CHECK ."','".$OGROUP_INPUT."','".$OYEAR
."','".$OSEAT ."'
,'2','2553','".$GPSEAT253 . "'
,'3','2553','".$GPSEAT353 . "'
,'1','2554','".$GPSEAT154 . "'
,'1','2555','".$GPSEAT155 . "'
,'2','2555','".$GPSEAT255 ."'
,'".$PPLID_CHECK."',  '".$NAME_CHECK."','".$SEX_INPUT;
$SQL =
$SQL."','".$SCHOOL_CHECK."','".$STUID10_CHECK."','".$BIRTHDATE_INPUT."','".$FYEAR_INPUT."','".$FS
TATE_INPUT."','".$GPAX_INPUT."',
'".$FCEN1."','".$FCEN2."','".$FCEN3."','".$FCEN4."','".$ADDRESS_INPUT."','".$TEL01_INPUT."','".$TEL02_IN
PUT."','".$DATE_KEY."','".$TIME_KEY."','".$TOTAL_PAY."','0','X')" OR DIE ("ADD ERROR ");

SELECT ADMREG FROM REGISTER WHERE SHSEQ='$SHSEQ_CHECK' AND KEYDATE='$DATE_KEY'
AND KEYTIME='$TIME_KEY';
SELECT SCHNAME, SCHPROV FROM SCHOOL WHERE SCHCODE ='$SCHOOL_CHECK';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN1';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN2';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN3';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN4';
UPDATE REGISTER SET FACROBAT='".$PDF."' WHERE ADMREG='$ADMREG';
```

PAGE5:

```
SELECT ADMREG,SHSEQ,PPLID FROM REGISTER WHERE ADMREG='$REGNO' AND PPLID='$PPLID';
```

PAGE6:

```
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN1';
```

```
SELECT
ADMREG,SHSEQ,PPLID,NAME,SEX,SCHOOL,STUID10,FYEAR,FCEN1,FCEN2,FCEN3,FCEN4,BILLTOTA
L,STATUSPAY,PAYDATE,PAYTIME  FROM REGISTER WHERE ADMREG='$REGNO' AND
PPLID='$PPLID';
SELECT SCHNAME, SCHPROV FROM SCHOOL WHERE SCHCODE ='$SCHOOL_INPUT';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN2';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN3';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN4';
```

PAGE7:

```
SELECT  OYEAR,OSEAT, GPSEAT253,GPSEAT353,GPSEAT154,GPSEAT155,GPSEAT255,
PPLID,NAME,SEX,SCHOOL,STUID10,FYEAR,FSTATE,GPAX,FCEN1,FCEN2,FCEN3,FCEN4,BILLTOTAL,A
DDRESS, TEL01,TEL02,KEYDATE,KEYTIME  FROM   REGISTER  WHERE ADMREG='$ADMREG' AND
PPLID='$PPLID' ";
SELECT SCHNAME, SCHPROV FROM SCHOOL WHERE SCHCODE ='$SCHOOL_INPUT';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN1';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN2';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN3';
SELECT NAME_FAC, NAME_UNIV FROM FACTNAME WHERE ID_FAC = '$FCEN4';
```

ประวัติผู้เขียนวิทยานิพนธ์

นางสาวสุวิมล ประทุม เกิดวันที่ 12 พฤษภาคม พ.ศ. 2527 สำเร็จการศึกษาระดับปริญญาบัณฑิต หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาการพัฒนาซอฟต์แวร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2549 และเข้าศึกษาระดับปริญญามหาบัณฑิต หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2552