

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง
ประสิทธิภาพของระบบจัดการฐานข้อมูลเชิงวัตถุสัมพันธ์
Performance Issues in Object Relational Database

โดย
นาย จักรพงศ์ แหยมงาม
นาย สิทธิโชค ฉัตรพ่องศรี

เลขหมู่.....
เลขทะเบียน.....
วัน,เดือน,ปี.....

7295ก

26 ส.ย. 2550

b. 111 15112

i.

ปฏิญานี้เป็นส่วนหนึ่งของการศึกษาค้นคว้าหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประสิทธิภาพของระบบจัดการฐานข้อมูลเชิงวัตถุสัมพันธ์

Performance Issues in Object Relational Database

โดย

นาย จักรพงศ์ แหยมงาม 47015316

นาย สิทธิโชค นัทรพ่องศรี 47015339

อาจารย์ที่ปรึกษา

อาจารย์ ศุภมิตร จิตตะขุโสธร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปีการศึกษา 2549

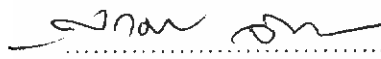
ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ประสิทธิภาพของระบบจัดการฐานข้อมูลเชิงวัตถุสัมพันธ์

ผู้จัดทำ

1. นาย จักรพงษ์ แหยมงาม
2. นาย สิทธิโชค ฉัตรผ่องศรี

.....อาจารย์ที่ปรึกษา

(รศ. ดร. สุกมิตร จิตตะยโสธร)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประสิทธิภาพของระบบจัดการฐานข้อมูลเชิงวัตถุสัมพันธ์

นาย จักรพงศ์ แหยมงาม

นาย สิทธิโชค ฉัตรผ่องศรี

อ. ศุภมิตร จิตะยโสธร

ปีการศึกษา 2549

บทคัดย่อ

โดยทั่วไปแล้วนั้นระบบจัดการฐานข้อมูลเดิมอันซึ่งประกอบไปด้วยส่วนต่าง ๆ ที่สำคัญ เพื่อที่จะให้ระบบฐานข้อมูลนั้นสามารถทำงานได้ แต่ระบบเดิมนั้นอาจจะไม่ดีที่สุดเพราะ รูปแบบและขนาดของการใช้งานที่แตกต่างกัน ซึ่งเป็นสาเหตุที่ทำให้ระบบฐานข้อมูลเดิมนั้นมีประสิทธิภาพที่แยกลง รวมถึงการใช้งานที่ไม่ได้เป็นการสนับสนุน การจัดการของระบบหรือ ข้อมูลเกิดขัดแย้งกัน ไม่ว่าจะด้วยสาเหตุใดก็ตาม นั้นหมายความว่าเราจะต้องรู้และเข้าใจโครงสร้างการทำงานและหลักการใช้งานทั่วไปที่สำคัญ รวมถึง การจัดการการฐานข้อมูลที่ถูกต้องเหมาะสม เพื่อช่วยย่นระยะเวลาของการทำงานและไม่เกิดผลกระทบกับข้อมูลที่ถูกเก็บไว้ในฐานข้อมูล รวมทั้งการปรับค่าพารามิเตอร์ต่าง ๆ ในระบบจากค่าเดิม(Default) ให้เหมาะสมกับปริมาณงานในแต่ละงานซึ่งจำเป็นอย่างยิ่งถ้าต้องการให้ระบบจัดการฐานข้อมูลมีประสิทธิภาพที่ดีที่สุด

ดังนั้นปัญญานิพนธ์ฉบับนี้ จึงได้ทำการรวบรวม โครงสร้างต่าง ๆ ทั่วไปของระบบจัดการฐานข้อมูล สถาปัตยกรรมต่าง ๆ ที่สำคัญทั้งหมดเพื่อที่จะให้เข้าใจถึงความสัมพันธ์โดยรวมของระบบ และควรที่จะจัดการฐานข้อมูลยังไงให้เหมาะสมกับปริมาณงานและทรัพยากรที่มีอยู่ในปัจจุบัน ได้อย่างมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Performance Issues in Object Relational Database

Jakapong Yamngam

Sittichok Chatpongsri

Suphamit Chittayasotorn

2006

Abstract

DBMS consists of many important components but the system is not the best because of there are different environments that are the reason why the first installed DBMS has low performance, does not fully support system and has data contradiction. DBMS has a performance tuning. No matter what reason we have to know structure, algorithm, and database management for decrease operation time and does not has data collision in database and adjust system parameters for specification environment

This thesis has been collected with important structure and architecture of DBMS which would allow the reader to have an idea about system. Also it provides the explanation of the performance tuning method which effects the efficiency of DBMS. Presently, we are using the performance tuning method to find the appropriate value suited with database's size and system's resource.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปฏิญานិพนธ์ฉบับนี้สามารถสำเร็จลุล่วงไปได้ด้วยดี ก็เพราะได้รับความช่วยเหลือ และการอนุเคราะห์จากหลาย ๆ ท่านด้วยกัน ซึ่งผู้จัดทำขอขอบพระคุณมา ณ ที่นี้

ขอขอบพระคุณ รศ.ดร. ศุภมิตร จิตตะโยธธ อาจารย์ที่ปรึกษาที่ได้ดูแลเอาใจใส่ตลอด ระยะเวลาการทำปฏิญานิพนธ์นี้ ให้คำแนะนำที่เป็นประโยชน์ ซึ่งแนะแนวทางการแก้ไขปัญหาต่าง ๆ ที่เกิดขึ้นระหว่าง การทำปฏิญานิพนธ์ พร้อมทั้งยังให้คำชี้แนะในการทำปฏิญานิพนธ์ฉบับนี้ด้วยดีมา โดยตลอด

ขอขอบคุณ ภาควิชาวิศวกรรมศาสตร์คอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง สำหรับสิ่งอำนวยความสะดวกมากมาย ที่ใช้สำหรับการทำวิจัยทั้งห้องทำงาน โต๊ะทำงาน อินเทอร์เน็ตสำหรับค้นคว้าหาข้อมูลต่าง ๆ ที่จำเป็นสำหรับการวิจัยนี้

ขอกราบพระคุณคณาจารย์ ภาควิชาวิศวกรรมศาสตร์คอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณ เพื่อน ๆ พี่ ๆ น้อง ๆ ที่คอยเอื้อเฟื้อ คอยช่วยเหลือด้านต่าง ๆ และคอยเป็นกำลังใจ ให้ผู้จัดทำมาโดยตลอด

ขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักรยิ่ง ซึ่ง ได้เลี้ยงดูผู้จัดทำมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอมา ในทุก ๆ ด้านอันหาที่เปรียบมิได้ ข้าพเจ้าขอระลึกในพระคุณอันสุดประมาณและขอกราบขอบพระคุณมา ณ ที่นี้

สุดท้ายนี้ ขอขอบพระคุณทุกท่านที่มีส่วนร่วมในการทำปฏิญานิพนธ์ฉบับนี้ ทั้งที่เอ่ยชื่อและ ไม่ได้เอ่ยชื่อก็ตาม ขอขอบพระคุณอีกครั้ง

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	1
บทคัดย่อภาษาอังกฤษ.....	2
กิตติกรรมประกาศ.....	3
สารบัญ.....	4-7
สารบัญรูปภาพ.....	8-9
สารบัญตาราง.....	10
บทที่ 1 บทนำ.....	11
1.1 ความสำคัญและที่มา.....	11
1.2 วัตถุประสงค์ของปริญญาานิพนธ์.....	11-12
1.3 ขอบเขตของปริญญาานิพนธ์.....	12
บทที่ 2 หลักการที่เกี่ยวข้องกับระบบจัดการฐานข้อมูล และการจัดการฐานข้อมูลโดยรวม.....	13
2.2.1 สถาปัตยกรรมต่าง ๆ ของระบบฐานข้อมูล.....	13-15
2.2.2 โครงสร้างของฐานข้อมูล.....	15
2.2.2.1 ดาต้าบีส็อก.....	15-16
2.2.2.2 เอ็กเทนส์.....	16
2.2.2.3 เซกเมนต์.....	16-17
2.2.2.4 เทเบิลส์ สเปซ.....	17
2.2.2.4.1 ชิสเต็ม เทเบิลสเปซ.....	18
2.2.2.4.2 นอน-ชิสเต็ม เทเบิล สเปซ.....	18-19
2.2.2.4.3 รูปแบบการจัดการพื้นที่ภายในเทเบิล สเปซ.....	19-20
2.2.2.5 ดาต้าไฟล์.....	20-21
2.2.2.6 ระดับชั้นต่าง ๆ ของระบบฐานข้อมูลออราเคิล.....	21-22
2.2.2.6.1 คอนเซ็ปชอลส์ เล็ฟวี่.....	21-22
2.2.2.6.2 อินเตอร์นอล เล็ฟวี่.....	22
2.2.2.6.3 เอ็กเซอร์นอล เล็ฟวี่.....	22
2.2.3 การปรับแต่งต่าง ๆ บนระบบฐานข้อมูล.....	22
2.2.4 การปรับแต่งระบบฐานข้อมูลและแสดงผล.....	22-23
2.2.5 หลักการอ่านเขียนข้อมูลจากฐานข้อมูล.....	23-24
2.2.6 โครงสร้างการจัดการไฟล์และเก็บข้อมูล.....	24-25

เอกสารนี้ 2.6 โครงสร้างการจัดการไฟล์และเก็บข้อมูล สืบจนเท่านั้น ไม่อนุญาตให้แก้ไขได้ ประโยชน์ 24-25 การค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
2.2.7 การปรับค่าพารามิเตอร์ต่าง ๆ ที่ใช้ในฐานข้อมูลของออราเคิล.....	25-26
บทที่ 3 การปรับแต่งประสิทธิภาพและจัดการฐานข้อมูล.....	27
3.1 Basic Index Principles.....	27
3.1.1 Basic Index Concept.....	27
3.1.2 Concatenated Indexes.....	28
3.1.3 The Oracle ROWID.....	28-29
3.1.4 Suppressing Indexes.....	29-30
3.1.5 Selectivity.....	30
3.1.6 Clustering Factor.....	31
3.1.7 Binary Height.....	31
3.1.8 Using Histograms.....	31
3.1.9 Fast Full Scans.....	32
3.1.10 Skip-Scan.....	32
3.1.11 B-tree Indexes.....	32-33
3.1.12 Bit-map Indexes.....	33-34
3.1.13 Hash Indexes.....	35
3.1.14 Index-Organized table.....	35
3.1.15 Function-Based Index.....	36-37
3.1.16 Partitioned Index.....	37
3.1.17 Local Indexes.....	37-38
3.1.18 Prefixed.....	38
3.1.19 Non prefixed.....	38
3.1.20 Global Indexes.....	39
3.2 การปรับแต่งฐานข้อมูลกับ Initialization parameters.....	39
3.2.1 เพิ่มประสิทธิภาพโดยการปรับตั้งค่า DB_CACHE_SIZE.....	39
3.2.2 ตั้งค่า DB_BLOCK_SIZE ให้สะท้อนถึงขนาดของการอ่านข้อมูล.....	39
3.2.3 DB_BLOCK_SIZE.....	39-40
3.2.4 DB_BLOCK_BUFFERS.....	40
3.2.5 DB_FILES.....	41
3.2.6 LOG_BUFFER.....	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
3.2.7 LOG_CHECKPOINT_INTERVAL.....	42
3.2.8 SHARED_POOL_SIZE.....	42-43
3.2.9 SORT_AREA_SIZE.....	43
3.2.10 SORT_AREA_RETAINED_SIZE.....	44
3.2.11 SQL_TRACE.....	44
3.2.12 LOG_CHECKPOINT_TIMEOUT.....	44
บทที่ 4 ผลการทดลอง.....	45
4.1 ทำการทดลองวัดอัตราการพบและไม่พบข้อมูลบนไลบรารี แคช และ ค่า ดิกชันนารี แคช.....	45
4.1.1 ขั้นตอนการทดลอง.....	45
4.1.2 ผลการทดลองวัดค่าอัตราการพบไม่พบข้อมูลบนไลบรารี แคช และค่า ดิกชันนารี แคช.....	46
4.2 การทดลองเปรียบเทียบการเข้าถึงข้อมูลแบบใช้อินเด็ก กับ ไม่ใช้อินเด็ก.....	47
4.2.1 ขั้นตอนการทดลอง.....	47-48
4.2.2 ผลการทดลองเปรียบเทียบอัตราการใช้ COST และ TIME.....	48
4.3 การทดลอง B-tree index และ Bit map index.....	49
4.3.1 ขั้นตอนการทดลองของ.....	49
4.3.1.1 ใช้คำสั่ง SET AUTOTRACE ON EXPLAIN.....	49
4.3.1.2 ทำการค้นหาข้อมูลในตาราง borrow ขณะที่ยังไม่มีอินเด็กซ์ที่ คอลัมน์ student_id.....	49
4.3.1.3 ทำการสร้าง B-tree อินเด็กซ์บนคอลัมน์ student_id.....	49-50
4.3.1.4 ทำการสร้าง Bitmap อินเด็กซ์บนคอลัมน์ student_id.....	50
4.3.1.5 ทำการค้นหาข้อมูลในตาราง student ขณะที่ยังไม่มีอินเด็กซ์ที่ คอลัมน์ en_name.....	51
4.3.1.6 ทำการสร้าง B-tree อินเด็กซ์บนคอลัมน์ en_name.....	51-52
4.3.1.7 ทำการสร้าง Bitmap อินเด็กซ์บนคอลัมน์ en_name.....	52
4.3.1.8 ทำการค้นหาข้อมูลในตาราง student ขณะที่ยังไม่มีอินเด็กซ์ที่ คอลัมน์ faculty_id.....	53
4.3.1.9 ทำการสร้าง B-tree อินเด็กซ์บนคอลัมน์ faculty_id.....	53-54
4.3.1.10 ทำการสร้าง Bitmap อินเด็กซ์บนคอลัมน์ faculty_id.....	54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
4.3.2 ผลการทดลองเปรียบเทียบค่า Cost จากแผนการรันที่ 1-9.....	55
4.4 การทดลองเรื่องการ Join.....	55
4.4.1 ขั้นตอนการทดลอง.....	55
4.4.1.1 ใช้คำสั่ง SET AUTOTRACE ON EXPLAIN.....	55-56
4.4.1.2 ทำการ Join ตาราง book กับตาราง borrow.....	56
บทที่ 5 สรุปผลการทดลอง.....	57
5.1 สรุปผลการทดลองเปรียบเทียบการเข้าถึงข้อมูลแบบใช้อินเด็ก กับ ไม่ใช้อินเด็ก.....	57
5.2 สรุปผลการทดลอง B-tree index และ Bit map index.....	57
5.3 สรุปผลการทดลอง การ Join.....	57
ภาคผนวก ก พารามิเตอร์ทั้งหมดของออร่าเคิสรุ่น 10g.....	58-61
ภาคผนวก ข วิว(V\$Views) ทั้งหมดของออร่าเคิสรุ่น 10g.....	62-67
ภาคผนวก ค ตารางที่ใช้ในการทดลอง.....	68-72
บรรณานุกรม.....	73



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปภาพ	หน้า
รูปที่ 2-1 ส่วนประกอบของ Oracle Instance.....	14
รูปที่ 2-2 ส่วนประกอบของ Oracle Database.....	14
รูปที่ 2-3 สถาปัตยกรรมทั้งหมดที่มีผลกับประสิทธิภาพของระบบ.....	15
รูปที่ 2-4 ความสัมพันธ์ของ คาด้านเบส, เทเบิลส์ สเปซ และ คาด้าไฟล์.....	17
รูปที่ 2-5 การประติดประต่อท่ามกลางระดับชั้น.....	21
รูปที่ 2-6 Step in query processing.....	23
รูปที่ 2-7 แสดงความสัมพันธ์ของ Physical และ Logical Structure ของฐานข้อมูล.....	24
รูปที่ 2-8 ALL Initialization Parameters.....	25
รูปที่ 2-9 ช่องว่าง และ ลิสบล็อกที่ใช้ใส่ค่าที่ต้องการจะปรับค่าพารามิเตอร์.....	26
รูปที่ 2-10 ปรับทาง SQL*Plus โดยใช้คำสั่ง ALTER SYSTEM.....	26
รูปที่ 3-1 Concatenated Indexes.....	28
รูปที่ 3-2 B-tree Index.....	33
รูปที่ 3-3 Bitmap Index.....	34
รูปที่ 3-4 Bitmap Index.....	34
รูปที่ 3-5 Bitmap Index.....	34
รูปที่ 3-6 Index-Organized Table.....	36
รูปที่ 3-7 Index-Organized Table.....	36
รูปที่ 3-8 prefixed partitioned index.....	38
รูปที่ 3-9 Non prefixed partitioned index.....	38
รูปที่ 3-10 ความสัมพันธ์ระหว่างDB_BLOCK_SIZE กับ DB_BLOCK_BUFFER.....	39
รูปที่ 4-1 อัตราการพบไม่พบข้อมูลในไลบรารี แคชที่ค่า 500,000 byte.....	45
รูปที่ 4-2 อัตราการพบไม่พบข้อมูลในคาด้า ดิกชันนารี แคชที่ค่า 500,000 byte.....	45
รูปที่ 4-3 อัตราการพบไม่พบข้อมูลในไลบรารี แคชที่ค่า 1,000,000 byte.....	46
รูปที่ 4-4 อัตราการพบไม่พบข้อมูลในคาด้า ดิกชันนารี แคชที่ค่า 1,000,000.....	46
รูปที่ 4-5 ข้อมูลทั้งหมดของเทเบิล LICEN.....	47
รูปที่ 4-6 แผนการรันเมื่อไม่เลือกใช้อินเด็กซ์.....	47
รูปที่ 4-7 แผนการรันเมื่อเลือกใช้อินเด็กซ์.....	48
รูปที่ 4-8 แผนการรันที่ 1.....	49
รูปที่ 4-9 แผนการรันที่ 2.....	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปภาพ	หน้า
รูปที่ 4-10 แผนการวันที่ 3.....	50
รูปที่ 4-11 แผนการวันที่ 4.....	51
รูปที่ 4-12 แผนการวันที่ 5.....	52
รูปที่ 4-13 แผนการวันที่ 6.....	52
รูปที่ 4-14 แผนการวันที่ 7.....	53
รูปที่ 4-15 แผนการวันที่ 8.....	54
รูปที่ 4-16 แผนการวันที่ 9.....	54
รูปที่ 4-17 Hash Join.....	56
รูปที่ 4-18 Hash Join.....	56



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 4-1 ผลการทดลองวัดค่าอัตราการพบไม่พบข้อมูล.....	46
ตารางที่ 4-2 ผลการทดลองเปรียบเทียบอัตราการใช้ COST และ TIME.....	48
ตารางที่ 4-3 แผนการรัน.....	55



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มา

ปัจจุบันฐานข้อมูลมีองค์ประกอบหลายอย่างที่มีส่วนเกี่ยวข้องเพื่อที่จะทำให้ฐานข้อมูลนั้นสามารถทำงานได้และส่วนต่าง ๆ เหล่านี้มีความสำคัญกับฐานข้อมูลแตกต่างกันไป ดังนั้นเราจึงต้องเข้าใจองค์ประกอบทุก ๆ ส่วนของฐานข้อมูลเพราะถ้าส่วนใดส่วนหนึ่งมีปัญหามันจะมีผลกระทบต่อประสิทธิภาพการทำงานของระบบฐานข้อมูล ฉะนั้นผู้ดูแลระบบจะต้องคอยตรวจสอบ ปรับแต่งประสิทธิภาพของระบบให้สามารถทำงานได้คืออยู่เสมอ ซึ่งออราเคิลเดิมนั้นนอกจากจะมีระบบจัดการที่ดีแล้วยังมีเครื่องมือ(Utilities) ต่าง ๆ เพื่อใช้ในการสนับสนุนการทำงานของออราเคิลเอง เช่น ออราเคิลเอนเตอร์ไพท์มานาเจอร์(Oracle Enterprise manager(OEM)), เอสคิวแอลพลัส(SQL*Plus), ออราเคิลมอนิเตอร์ริง(Oracle Moniterring), อะเลิทแคตทากอรี(Alerts Category), ไอเอสคิวแอลพลัส(i SQL*Plus) เป็นต้น

เนื่องจากระบบคอมพิวเตอร์ ที่มีจำนวนทรัพยากรของระบบไม่เท่ากันดังนั้น ออราเคิลจึงให้ความสามารถ ในการปรับแต่งระบบฐานข้อมูล เพื่อให้เหมาะสมกับขนาดของทรัพยากรที่มีอยู่ทั้งระดับลอจิกคอล(logical) ให้สัมพันธ์กับระดับฟิซิคัล(Physical) ได้อย่างถูกต้องและพยายามไม่ให้เกิดปัญหาบ่อย ๆ กับระบบเพื่อให้ระบบมีประสิทธิภาพ, รวดเร็ว, และยืดหยุ่นที่สุด แต่อย่างไรก็ตามบุคคลที่จะเข้าไปทำการจัดการฐานข้อมูลด้านประสิทธิภาพได้นั้นจะต้องมีความรู้ละเอียดความเข้าใจเกี่ยวกับสถาปัตยกรรมของระบบคอมพิวเตอร์และระบบจัดการฐานข้อมูลด้วย

1.2 วัตถุประสงค์ของปริญญานิพนธ์

- 1.2.1 เพื่อศึกษาและทำความเข้าใจถึงการทำงานของระบบฐานข้อมูลของออราเคิล รวมถึงโครงสร้างโดยทั่วไปของระบบฐานข้อมูล เช่น โครงสร้างฐานข้อมูล(Database Structure) และสถาปัตยกรรมของระบบ (System Architecture) ที่มีผลต่อประสิทธิภาพการทำงานของระบบฐานข้อมูลของออราเคิล
- 1.2.2 เพื่อให้ทราบถึงพารามิเตอร์และปัจจัยต่าง ๆ ที่มีผลต่อประสิทธิภาพการทำงานของระบบฐานข้อมูลของออราเคิล
- 1.2.3 เพื่อให้ทราบถึงคำสั่งต่าง ๆ ที่สำคัญสำหรับใช้ปรับแต่งประสิทธิภาพและจัดการฐานข้อมูลสำหรับนักศึกษาหรือบุคคลทั่วไปที่สนใจที่จะนำไปดูเพื่อประกอบการทำงานในชีวิตจริงหรือการเรียนการรู้ด้วยตัวเองสำหรับผู้เริ่มต้นกับการจัดการฐานข้อมูลของออราเคิลค่าเบส10g ให้มีประสิทธิภาพดีที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1.2.4 เพื่อศึกษาและทำความเข้าใจว่าทำอย่างไรระบบฐานข้อมูลของออราเคิลจึงจะมีประสิทธิภาพดี ที่สุดและมีความเหมาะสมที่สุดกับปริมาณงานในแต่ละงานซึ่งแปรผันตรงกับระบบปฏิบัติการ (Window Xp) และทรัพยากรของเครื่องแต่ละเครื่องที่มีอยู่เดิม

1.3 ขอบเขตของปริญญานิพนธ์

ระบบฐานข้อมูลที่ใช้ในการศึกษาคือออราเคิลดาต้าเบส 10g ที่สามารถปรับเปลี่ยนค่าพารามิเตอร์ เพื่อปรับแต่งประสิทธิภาพของระบบได้ ทั้งแบบ User Interface ผ่าน web Browser หรือ ใช้ Command Line ติดต่อกับฐานข้อมูลผ่านทาง SQL Plus และสามารถดูผลการปรับได้จาก External view ผ่าน GUI ของออราเคิล ส่วนด้านการทดลองใช้ออราเคิล Enterprise Manager Database Control (OEM) เพื่อดูผลการทดลองและกราฟต่าง ๆ ที่บอกสถานะของประสิทธิภาพ เพื่อนำมาพิจารณาและดูแนวโน้มของการทดลองว่าจะมีโอกาสเปลี่ยนแปลงไปมากน้อยแค่ไหน ส่วนการที่เราต้องการที่จะดูค่าจริง ๆ หรือรูปแบบการทดลองที่เราได้ออกแบบไว้ว่าใช้หลักการและทฤษฎีแบบไหนตรงตามที่เรต้องการหรือไม่ เพื่อที่จะได้นำมาวัดและเปรียบเทียบกันได้นั้นสามารถทำได้โดยการ Execution Plan งานผลการทดลองออกมาดูและนำผลที่ได้มาตรวจสอบกับทฤษฎีที่ใช้ว่าแบบไหนที่มีประสิทธิภาพดีกว่ากันทำอย่างไรให้ อะไรถึงจะเร็วและมีประสิทธิภาพดีที่สุดกับระบบฐานข้อมูลแล้วสิ่งไหนที่ไม่ควรหรือทำแล้วจะทำให้ ประสิทธิภาพของฐานข้อมูลแยกลงช้าลงมาก ๆ หรือว่าการทำแบบนี้อาจจะไม่เร็วที่สุดแต่ก็ควรทำเพราะ มันเหมาะสมกับปริมาณงานที่เราทำอยู่บ่อย ๆ ซึ่งควรจะสามารถปรับแต่งให้ระบบฐานข้อมูลที่ใช้อยู่มีความยืดหยุ่นกับงานในแต่ละงานมากที่สุดถึงจะเป็นสิ่งที่ดีที่สุดสำหรับประสิทธิภาพโดยรวมทั้งหมด ของระบบซึ่งเป็นจุดมุ่งหมายที่สำคัญที่สุดของปริญญานิพนธ์เล่มนี้ ซึ่งทางคณะผู้จัดทำหวังว่ามันจะเป็น ประโยชน์ให้กับนักศึกษาหรือบุคคลทั่วไปที่สนใจด้านการจัดการฐานข้อมูลให้มีประสิทธิภาพไม่มาก ก็น้อย โดยหลักทั่ว ๆ ไป เราได้เน้นไปที่การทำการปรับปรุงแก้ไขและพัฒนาให้การทำงานและการ จัดการงานของระบบฐานข้อมูลออราเคิลดาต้าเบส 10g เกิดความเหมาะสมและมีประสิทธิภาพดีที่ สุดบนระบบปฏิบัติการวินโดวส์เอ็กซ์พี (Window Xp) และมีทรัพยากรให้ใช้ตามอัตราภาพเท่าที่มีอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

หลักการที่เกี่ยวข้องกับระบบจัดการฐานข้อมูล และการจัดการฐานข้อมูลโดยรวม

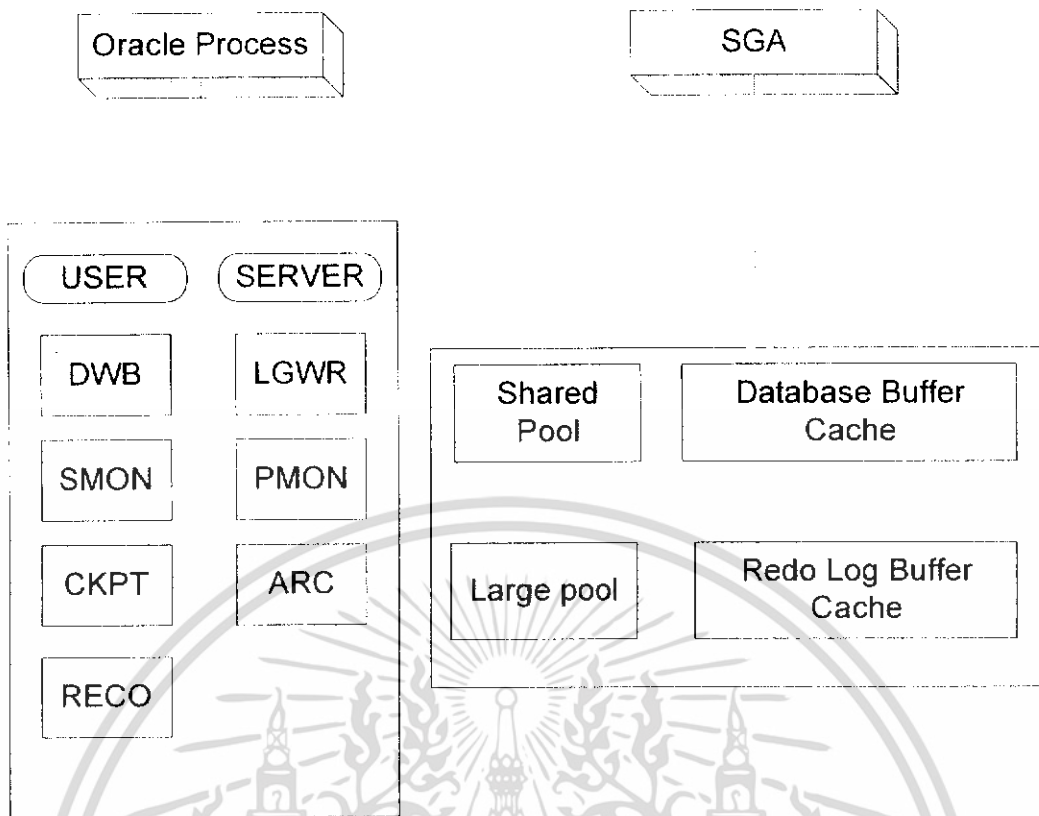
2.2.1 สถาปัตยกรรมต่าง ๆ ของระบบฐานข้อมูล (System Architecture) ประกอบไปด้วย ส่วนประกอบหลัก 2 ส่วนคือ Oracle Instance และ Oracle Database

โดย **Oracle Instance** คือหน่วยความจำที่ใช้เก็บข้อมูลและความคุมการทำงานของฐานข้อมูลของออราเคิลที่เรียกว่า SGA (System Global Area) ร่วมกับ Oracle Process ที่ใช้ในการทำงานของออราเคิล

SGA และ Oracle Process ที่รวมกันเป็น Oracle Instance ยังประกอบไปด้วยส่วนย่อย ๆ อีกหลายส่วน โดย SGA นั้นประกอบไปด้วยหน่วยความจำที่สำคัญคือ Shard Pool, Database Buffer Cache, RedoLog Buffer และ large pool ส่วน Oracle Process นั้นประกอบไปด้วย Process ต่าง ๆ ที่ใช้ในการทำงานของออราเคิล ได้แก่

- ยูสเซอร์ โพรเซส (User Process)
- โพรเซสของตัวให้บริการฐานข้อมูล (Server Process)
- ดาต้าเบส ไรต์เตอร์ โพรเซส (Database Writer Process : DBW)
- ล็อก ไรต์เตอร์ (Log Writer Process : LGWR)
- ซีสเต็มมอนิเตอร์ (System Monitor : SMON)
- โพรเซส มอนิเตอร์ (Process Monitor : PMON)
- เช็คพอยท์ (Checkpoint Process : CKPT)
- รีคอบเวอร์เรอร์ (Recovered Process : RECO)
- อาร์ไคิวเวอร์ (Archiver Process : ARC)

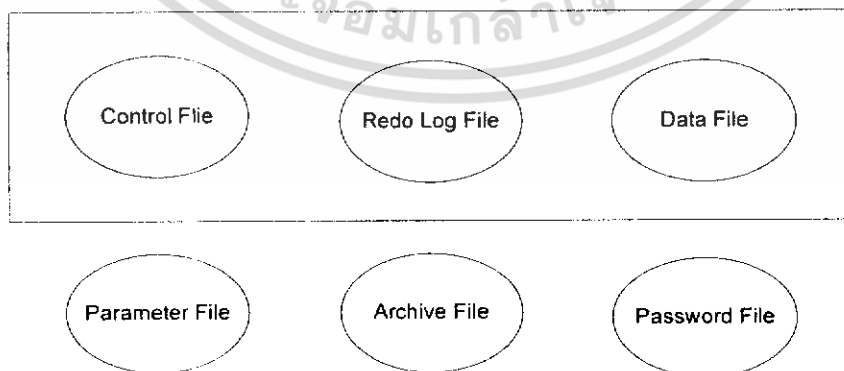
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-1 ส่วนประกอบของ Oracle Instance

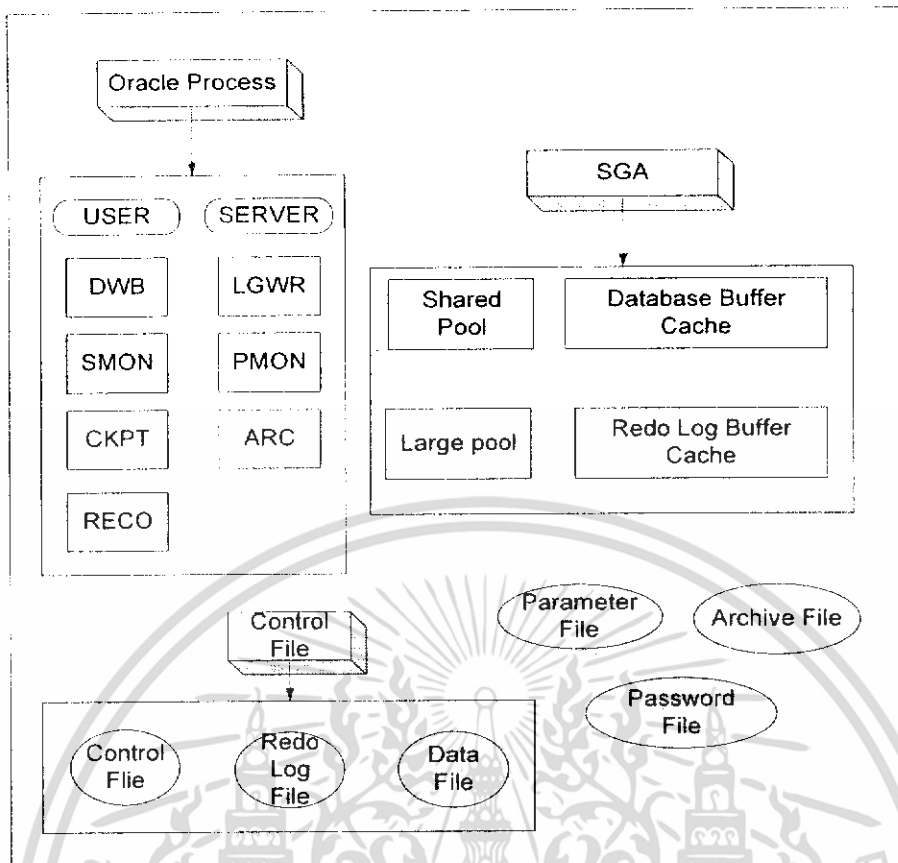
ส่วน **Oracle Database** คือ กลุ่มข้อมูลที่ถูกรวบรวมไว้ด้วยกัน โดยจัดเก็บลงเป็น ไฟล์ข้อมูลที่เป็น Physical File ซึ่งเป็นข้อมูลที่เก็บอยู่ในไฟล์เหล่านี้สามารถที่จะเรียกขึ้นมาดูหรือเปลี่ยนแปลงค่าได้

ไฟล์ต่าง ๆ ที่รวมกันเป็น Oracle Database นั้นมีอยู่ 3 ประเภทคือ Control File, Redo Log File และ Data File ซึ่งแต่ละ ไฟล์ก็มีหน้าที่แตกต่างกัน นอกจากนี้ไฟล์ทั้งสาม 3 ประเภทที่ ถึงแม้ว่าจะไม่ใช่ ส่วนประกอบใน Oracle Database แต่ก็มีความสำคัญในการทำงานของฐานข้อมูล Oracle เช่นเดียวกัน ได้แก่ Parameter File, Archive File, และ Password File โดยรายละเอียดของแต่ละส่วนประกอบนั้นจะ แสดงในรูปที่ 2-2



รูปที่ 2-2 ส่วนประกอบของ Oracle Database

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-3 สถาปัตยกรรมทั้งหมดที่มีผลกับประสิทธิภาพของระบบ

2.2.2 โครงสร้างของฐานข้อมูล(Database structure) เป็นการศึกษาโดยทั่ว ๆ ไปของระบบฐานข้อมูล(Database System) ได้แก่

ดาต้า บล็อก(Data Block)

เอ็กเทนต์(Extent)

เซกเมนต์(Segment)

เทเบิลสเปซ(Tablespace)

ระดับชั้นของฐานข้อมูล(Three Level of a Database)

2.2.2.1 ดาต้าบล็อก(Data Block)

การเก็บข้อมูลในฐานข้อมูลนั้น พื้นที่ที่ใช้เก็บข้อมูลคือ เทเบิลสเปซ (Tablespace) และใน เทเบิลสเปซเองแท้ที่จริงแล้วจะประกอบไปด้วย ดาต้าไฟล์ (DataFile) อย่างน้อย หนึ่งไฟล์รวมกันอยู่ ดังนั้นข้อมูลจริง ๆ จึงถูกเก็บอยู่ใน ดาต้าไฟล์ นั้นเอง

ในดาต้าไฟล์ จะแบ่งออกเป็นส่วนย่อย ๆ ที่เรียกว่าบล็อก(Block) โดยบล็อกแต่ละบล็อก จะมีขนาดเท่ากัน แล้วแต่ที่เราได้กำหนดตอนสร้างฐานข้อมูลไว้อย่างไร เช่น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2k, 4k, หรือ 8k บล็อกในค่าไฟล์ นี้จะเป็นพื้นที่ที่ใช้เก็บข้อมูลส่วนที่น้อยที่สุดของออบเจกต์ ซึ่งเราเรียกว่า ค่าบล็อก(Data Block) ค่าบล็อกเป็น ลอจิคอลสตรัคเจอร์(Logical Structure) ส่วนที่เล็กที่สุดของฐานข้อมูล

2.2.2.2 เอ็กเทนต์(Extent)

เอ็กเทนต์(Extent) คือ ลอจิคอลสตรัคเจอร์(Logical Structure) ที่มีขนาดใหญ่กว่าค่าบล็อกโดย เอ็กเทนต์ คือ ค่าบล็อกหลายๆ บล็อกที่มีพื้นที่ต่อเนื่องกันรวมเข้าด้วยกัน และ เอ็กเทนต์ นี้จะเป็นหน่วยย่อยที่สุดที่ออบเจกต์สามารถจองพื้นที่ใช้งานในเทเบิลสเปซ(Tablespace) ของฐานข้อมูลได้ ดังนั้นการเก็บข้อมูลของแต่ละออบเจกต์จึงเก็บอยู่ใน เอ็กเทนต์(Extent) อย่างน้อย หนึ่ง เอ็กเทนต์(Extent) เสมอ

2.2.2.3 เซกเมนต์(Segment)

เซกเมนต์(Segment) คือ ลอจิคอลสตรัคเจอร์(Logical Structure) ที่มีขนาดรองลงมาจาก เทเบิลสเปซ(Tablespace) โดยเราจะเรียกออบเจกต์ที่ต้องการใช้พื้นที่ในการเก็บข้อมูลว่า เซกเมนต์(Segment) ดังนั้นจึงสามารถกล่าวได้ว่าขนาดของเซกเมนต์คือพื้นที่ทั้งหมดที่ออบเจกต์จองไว้เพื่อใช้งาน

เราจะแบ่งประเภทของ เซกเมนต์(Segment) ตามประเภทของออบเจกต์ เช่น เราสามารถเรียกดาร์จเอ็มพลอยยี (Employee) ว่า เซกเมนต์เอ็มพลอยยี (Segment Employee) โดยมีประเภทของ เซกเมนต์(Segment) เป็นดาร์จ และถ้า เซกเมนต์เอ็มพลอยยี (Segment Employee) ประกอบด้วย 5 เอ็กเทนต์ ซึ่งแต่ละเอ็กเทนต์ มีขนาดเท่ากับ 1 เมกกะไบต์ ดาร์จเอ็มพลอยยี (Employee) ก็จะใช้พื้นที่ในเทเบิลสเปซ(Tablespace) เท่ากับ 5 เมกกะไบต์

ประเภทของ เซกเมนต์(Segment) ในฐานข้อมูลแบ่งออกได้เป็น 4 ประเภท

- ค่าเซกเมนต์(Data Segment) เป็น เซกเมนต์ ที่ใช้เก็บข้อมูลในฐานข้อมูล อย่างเช่น ดาร์จหรือ คลัสเตอร์เทเบิล(Clustered table) ต่างก็เป็นค่าเซกเมนต์(Data Segment) เพื่อเก็บข้อมูล
- อินเด็กซ์เซกเมนต์(Index Segment) เป็น เซกเมนต์ที่ใช้เก็บข้อมูลประเภทอินเด็กซ์ที่สร้างขึ้นในฐานข้อมูล
- โรลแบ็กเซกเมนต์(Rollback Segment) เป็น เซกเมนต์ ที่ใช้เก็บข้อมูลก่อนที่ จะมีการเปลี่ยนแปลงกับฐานข้อมูลจริง ๆ เพื่อให้ยูสเซอร์สามารถ โรลแบ็ก(Rollback) การ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้กับการแข่งขันเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

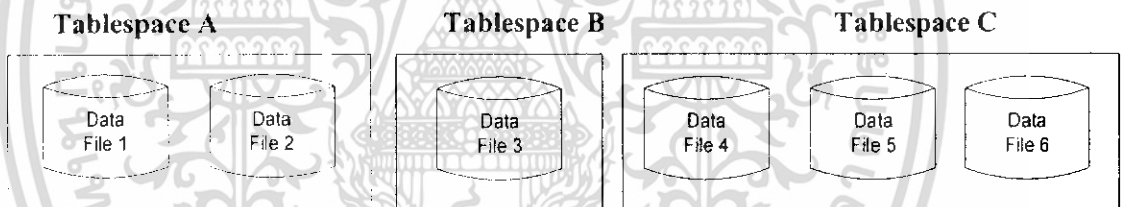
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

- เพิ่มโพลาริเซ็กเมนต์(Temporary segment) เป็น เซกเมนต์ ที่ใช้เก็บค่าข้อมูลชั่วคราว โดยถูกสร้างมาจากออร์วาคิลเองเพื่อใช้ไมการทำงานต่าง ๆ เช่นการเรียงลำดับข้อมูล ซึ่งหลังจากที่ทำงานที่ต้องการเสร็จแล้วพื้นที่นี้จะถูกคืนกลับให้กับระบบ

2.2.2.4 เทเบิลสเปซ(Tablespace)

เทเบิลสเปซ คือ ลอจิคอลสตักเจอร์(Logical Structure) ประเภทหนึ่งของฐานข้อมูลที่ใช้เป็นพื้นที่ในการเก็บข้อมูลต่าง ๆ ของฐานข้อมูล โดยในฐานข้อมูลหนึ่ง ๆ จะประกอบด้วย เทเบิลสเปซหลายตัวรวมเข้าด้วยกัน

เทเบิลสเปซ ถูกสร้างอยู่บนดาต้าไฟล์ (Data File) ที่เป็น ฟิสิคัล ไฟล์ ซิสเต็ม (Physical File System) จริงในระบบ แต่ละ เทเบิลสเปซ อาจจะประกอบด้วยหลาย ๆ ไฟล์รวมกันแต่อย่างน้อยที่สุดแล้วใน 1 เทเบิลสเปซ จะต้องมิดาต้าไฟล์(Data File) อยู่ 1 ไฟล์เสมอ ขนาดของ เทเบิลสเปซ ก็คือขนาดของดาต้าไฟล์ที่ประกอบเป็น เทเบิลสเปซ รวมกัน และขนาดของ เทเบิลสเปซ ทั้งหมดรวมกันก็จะกลายเป็นขนาดของฐานข้อมูลตั้งรูป



รูป 2-4 ความสัมพันธ์ของ ดาต้าเบส, เทเบิลสเปซ และ ดาต้าไฟล์

ในการสร้างเทเบิลสเปซ(Tablespace) เราจะต้องกำหนดชื่อและขนาดของไฟล์ที่จะประกอบเป็นเทเบิลสเปซ(Tablespace) ซึ่งเป็นส่วนที่ขาดไม่ได้ นอกจากนี้สิ่งที่เราควรจะต้องกำหนดอีกอย่างหนึ่งก็คือ ค่าดีฟอลต์(Default) ในการจองพื้นที่การใช้งานของออบเจกต์ในเทเบิลสเปซ(Tablespace) นี้ซึ่งเรียกว่าค่า ดีฟอลต์สโตเรจ(Default Storage) หากยูสเซอร์มีการสร้างออบเจกต์ เช่น ตารางอินเด็กซ์ในเทเบิลสเปซ(Tablespace) โดยที่ไม่ได้กำหนดค่า การจองพื้นที่ สโตเรจ(Storage) ให้ออบเจกต์(Object) เหล่านั้น การสร้างออบเจกต์จะใช้ค่าการจองพื้นที่ที่เป็นค่าดีฟอลต์(Default)ใน เทเบิลสเปซ(Tablespace) แทน

72950

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้จัดการฐานข้อมูลสามารถที่จะทำการใช้เทเบิล สเปซในการทำในการทำสิ่งต่าง ๆ ดังต่อไปนี้

- ควบคุมการจับจองที่ว่างสำหรับข้อมูลของข้อมูลในดิสก์
- กำหนดส่วนแบ่งดิสก์(Disk Quota) สำหรับผู้ใช้งานระบบฐานข้อมูลของออราเคิล
- ควบคุมการใช้งานของข้อมูลในเทเบิลสเปซโดยทำการตั้งให้เป็นออนไลน์ (Online) หรือ ออฟไลน์(Offline)
- กระทำการสำรองข้อมูล(Backup) หรือการคืนสู่สภาพเดิม(Recovery) ของฐานข้อมูลเฉพาะบางส่วนได้
- ทำการจับจองหน่วยความจำสำรองข้อมูล(Data Storage) ข้ามอุปกรณ์สำหรับการปรับปรุงประสิทธิภาพ

2.2.2.4.1 ซิสเต็ม เทเบิลสเปซ(System tablespace)

จัดเป็นเทเบิล สเปซชนิดหนึ่งซึ่งทุก ๆ ฐานข้อมูลของออราเคิลจะมีเทเบิล สเปซที่ชื่อว่าซิสเต็ม(System) ซึ่งออราเคิลจะทำการสร้างเทเบิล สเปซ ชนิดนี้ขึ้น เมื่อมีการสร้างฐานข้อมูลโดยอัตโนมัติ ซิสเต็ม เทเบิล สเปซ จะทำการบรรจุตารางของดาต้าดิคชันนารี(Data Dictionary) สำหรับฐานข้อมูลทั้งหมด ในกรณีที่ฐานข้อมูลมีขนาดเล็ก อาจไม่จำเป็นที่จะต้องมีการสร้างเทเบิล สเปซ ใหม่ขึ้นมา แต่ในกรณีที่ขนาดของฐานข้อมูลมีขนาดใหญ่ อาจจำเป็นที่จะต้องมีการสร้างเทเบิล สเปซ ขึ้นใหม่สำหรับผู้ใช้งานในระบบ เพราะต้องจับจองเนื้อที่ ที่เหมาะสมให้กับงานที่ต้องการได้ ซึ่งจะเป็นการปรับปรุงประสิทธิภาพของการเข้าถึงข้อมูลบนฐานข้อมูลได้เป็นอย่างดี(ซิสเต็ม เทเบิลสเปซควรมีการออนไลน์ตลอดเวลาและ ในซิสเต็ม เทเบิล สเปซ ไม่ควรจะให้เกิดการ Fragment ขึ้น เพื่อประสิทธิภาพที่ดีในการทำงานของฐานข้อมูลเอง)

2.2.2.4.2 นอน-ซิสเต็ม เทเบิล สเปซ(Non - System tablespace)

เป็นเทเบิล สเปซ ที่ใช้เก็บข้อมูลอื่น ๆ ของฐานข้อมูล เช่น ข้อมูลของยูสเซอร์ต่าง ๆ รวมทั้งพื้นที่ที่ใช้งานสำหรับการทำงานบางอย่างของฐานข้อมูล เช่น การเรียงลำดับข้อมูล นอน-ซิสเต็ม เทเบิล สเปซ แบ่งออกได้ดังนี้

- โรลแบ็ก เทเบิลสเปซ(Rollback Tablespace)
เป็นเทเบิลสเปซ ที่ใช้เก็บ โรลแบ็กเซ็กเมนต์(Rollback segment)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **เทมโพลารี เทเบิล สเปซ(Temporary Tablespace)**

เป็นเทเบิลสเปซที่ใช้ งานชั่วคราวสำหรับการทำงานของฐานข้อมูลบางอย่าง เช่น การเรียกดูข้อมูลจากฐานข้อมูลที่ต้องมีการเรียงลำดับข้อมูล ก่อนหลัง หากพื้นที่ ที่ใช้ในการเรียงลำดับข้อมูลในหน่วยความจำไม่ เพียงพอกับการใช้งานแล้ว ออราเคิล จะใช้พื้นที่ใน เทม โพลารี เทเบิล สเปซ (Temporary Tablespace) เพิ่มเติมเพื่อช่วยในการเรียงลำดับข้อมูล

- **ดาต้า เทเบิล สเปซ(Data Tablespace)**

เป็นเทเบิลสเปซที่ใช้เก็บข้อมูลของอ็อบเจกต์ต่าง ๆ ของยูสเซอร์ใน ฐานข้อมูล เช่น ตารางข้อมูล

- **อินเด็กซ์ เทเบิล สเปซ(Index Tablespace)**

เป็น เทเบิล สเปซที่ใช้ สำหรับเก็บอินเด็กซ์(Index) ซึ่งควรจะแยกพื้นที่การ เก็บอินเด็กซ์(Index) ออกจากเทเบิล สเปซ ที่เก็บข้อมูล

2.2.2.4.3 รูปแบบการจัดการพื้นที่ภายในเทเบิล สเปซ(Tablespace)

พื้นที่ภายใน เทเบิล สเปซ จะแบ่งออกเป็น ส่วน ๆ เรียกว่า เอ็กเทนต์(Extent) ที่ประกอบด้วยพื้นที่ว่างที่ยังไม่ใช้เก็บข้อมูล และพื้นที่ ที่มีข้อมูลเก็บอยู่ แล้ว

วิธีการจัดการพื้นที่ในเทเบิล สเปซ เพื่อจัดการใช้งาน เอ็กเทนต์ ใน เทเบิล สเปซ นั้นสามารถแบ่งออกได้ 2 วิธีคือ

1. การจัดการดิกชันนารี (Dictionary Management)

เป็นการจัดการพื้นที่การใช้งาน เทเบิล สเปซ ผ่าน ดาต้าดิกชันนารี (Data dictionary) ของฐานข้อมูล โดยข้อมูลที่เป็นที่ว่างในเทเบิล สเปซ จะถูก จัดเก็บในดาต้าดิกชันนารีของออราเคิลและเมื่อมีการเปลี่ยนแปลงพื้นที่ภายใน เทเบิล สเปซ เช่นจำนวนพื้นที่ว่างในเทเบิล สเปซ เพิ่มขึ้นหรือลดลงก็จะมีการ เปลี่ยนแปลงข้อมูลเหล่านี้ในดาต้าดิกชันนารี ด้วยทุกครั้งเพื่อให้สามารถทราบ ได้ว่ามีพื้นที่เหลืออยู่เท่าไรในแต่ละเทเบิล สเปซ

2. การจัดการแบบ โลกคอล(Locally Management)

เป็นการจัดการพื้นที่การใช้งานภายใน เทเบิล สเปซ โดยไม่ผ่านดาต้า ดิกชันนารี(Data dictionary) นั่นก็คือข้อมูลฟรีลิส(Free List) จะถูกจัดเก็บใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ดูแลระบบเห็นประโยชน์ในการค้า ไม่ว่ากรรมใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของบิตแมพอินเด็กซ์(Bit-Map Index) เช่น 10011111.... โดย 1 แทนกลุ่มของบล็อก(Block) ข้อมูลที่ว่าง วิธีนี้มีข้อดีคือ เมื่อต้องการหาที่ว่างภายในเทเบิล สเปนซ์ สามารถทำการแมพ(Map) ได้จากบิตแมพอินเด็กซ์โดยตรง ไม่ต้องค้นหาจาก คาด้าดิกชันนารี(Data dictionary) ทำให้มีความรวดเร็วในการทำงานมากยิ่งขึ้น นอกจากนี้เมื่อพื้นที่ภายในเกิดการเปลี่ยนแปลงอรรถาณ จะทำการเปลี่ยนค่าของ บิตแมพ(Bit-Map) นี้จาก 1 เป็น 0 หรือ 0 เป็น 1 ได้ทันทีโดยไม่ต้องผ่านคาด้าดิกชันนารี

2.2.2.5 คาด้าไฟล์(Data Files)

เมื่อคาด้าไฟล์ถูกสร้างขึ้นสำหรับเทเบิล สเปนซ์, อรรถาณ จะทำการสร้างแฟ้มข้อมูล(File) โดยทำการจับจองที่ว่างบนดิสก์ที่ระบุบวกกับ โอเวอร์เฮด(Overhead) ที่จำเป็นในการเก็บส่วนหัวของแฟ้มข้อมูล(File Header) เมื่อคาด้าไฟล์ถูกสร้างขึ้น ระบบปฏิบัติการจะทำการรับผิดชอบในเรื่องดังต่อไปนี้

- ลบข้อมูลเก่า ๆ ทิ้งไป
- การอนุญาตให้เข้าสู่ระบบ(Authorization) ก่อนที่จะทำการจับจองที่ว่างบนดิสก์ ถ้าคาด้าไฟล์ที่จะทำการสร้างมีขนาดใหญ่มาก ๆ ก็จะทำให้ใช้เวลานาน

คาด้าไฟล์(DataFile Content)

คาด้าไฟล์ที่ถูกสร้างขึ้น ที่ว่างบนดิสก์ที่ถูกจับจองจะยังไม่บรรจุข้อมูลอะไรเลย อย่างไรก็ตาม อรรถาณ ก็ทำการจับจองที่ว่างเอาไว้ สำหรับเก็บข้อมูลที่จะเพิ่มขึ้นในอนาคตในลักษณะที่สัมพันธ์กับเทเบิล สเปนซ์

ขนาดของคาด้าไฟล์(DataFile Size)

เราสามารถทำการเปลี่ยนขนาดของคาด้าไฟล์ หลังจากที่มีมันถูกสร้างขึ้นแล้ว หรือใช้วิธีระบุขนาดของคาด้าไฟล์ให้เป็นลักษณะการขยายแบบไดนามิก(Dynamic)

ออฟไลน์ คาด้าไฟล์(Offline DataFile)

เราสามารถที่จะทำการติดตั้งให้เทเบิล สเปนซ์ออนไลน์หรือออฟไลน์เมื่อใดก็ได้ แต่กระนั้นก็จะทำให้คาด้าไฟล์ถูกทำเป็นออนไลน์ตามไปด้วย หรือออฟไลน์ตามไปด้วยอย่างไรก็ดีเราสามารถทำการตั้งออฟไลน์เป็นคาด้าไฟล์ได้ ในกระบวนการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บางอย่าง คาด้าไฟล์จะถูกลบออกออนไลน์โดยอัตโนมัติ กระบวนการเหล่านี้ได้แก่การกลับคืนสู่สภาพเดิมของฐานข้อมูล(Database Recovery)

2.2.2.6 ระดับชั้นต่าง ๆ ของระบบฐานข้อมูลออราเคิล

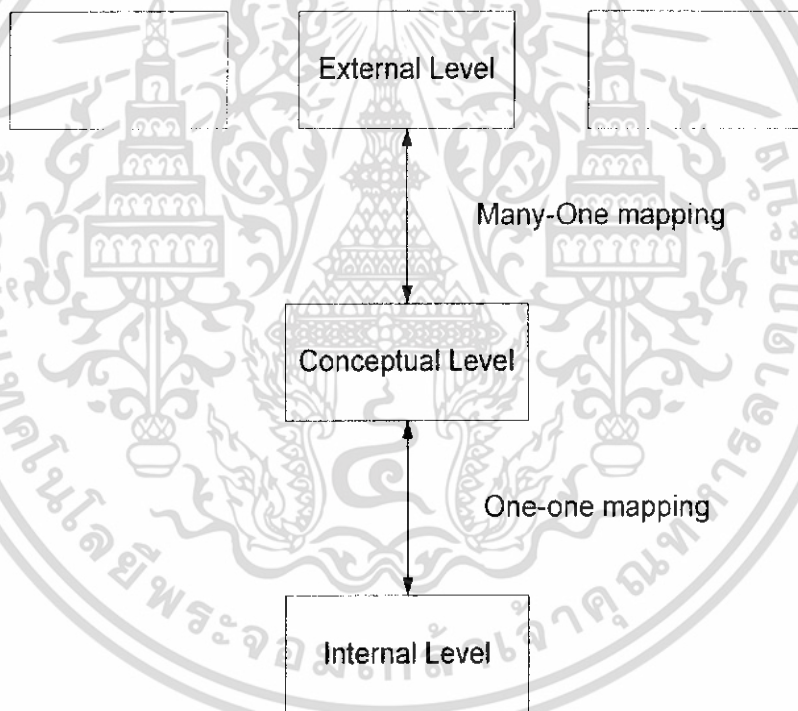
สถาปัตยกรรมของฐานข้อมูลออราเคิล ประกอบด้วย 3 ความแตกต่างแต่นั้น ถูกทำให้มีความสัมพันธ์ร่วมกันในแต่ละระดับคือ

2.2.2.5.1 คอนเซ็ปชอลว์ เล็ฟวีล(Conceptual Level)

2.2.2.5.2 อินเทอร์เน็ต เล็ฟวีล(Internal Level)

2.2.2.5.3 เอ็กเทอร์เน็ต เล็ฟวีล(External Level)

ฐานข้อมูลจะทำงานได้ก็จาก 3 ระดับนี้ และอาจจะมีส่วนทำให้ประสิทธิภาพแย่งถ้าอันใดอันหนึ่งเกิดข้อบกพร่องขึ้นระหว่างสามระดับนี้ ดังรูปที่ 2-5



รูปที่ 2- 5 การประติดประต่อท่ามกลางระดับชั้น

2.2.2.6.1 คอนเซ็ปชอลว์ เล็ฟวีล(Conceptual Level) แสดงให้เห็นว่าข้อมูลข่าวสารทั้งหมดที่บรรจุอยู่ ในฐานข้อมูลมีความสำคัญที่สุดในโปรแกรมฐานข้อมูล ระดับ คอนเซ็ปชอลว์ ถูกกำหนดไว้ที่นี้รวมถึงข้อมูลข่าวสารสัมพันธ์กับโครงสร้างทั้งหมดเกี่ยวกับวัตถุข้อมูลทั้งหมดที่เป็นของฐานข้อมูล จากจุดยืนการออกแบบทางตรรกะของมันรวมถึงนิยามของ เทเบิล, ดัชนี, คอนสเตรนท, วิว, ทริกเกอร์, โพรซีเจอร์, ฟังก์ชัน, แพกเกจ, ซี

โนนิม ๆ จากจุดยืนการออกแบบทางกายภาพที่มันบรรจุการเจาะจงของ เทเบิลสเปซ, คาด้า เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการเผยแพร่เพื่อการศึกษาเท่านั้น มิใช่เอกสารการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟล์, และอนุประ โยคที่ใช้บันทึกของ ตารางทั้งหมด และ ดัชนี เพื่อข้อสรุป, ระดับ คอนเซ็ปชอลว์ มันจัดเตรียมรายละเอียดทั้งหมดสำหรับอิมพลีเมนต์ฐานข้อมูล

2.2.2.6.2 อินเทอร์เน็ต เลฟเว็ล (Internal Level) ถูกเรียกว่าระดับทางกายภาพคือ ใกล้เคียงที่สุดที่ใช้บันทึกทางกายภาพที่ซึ่งมันบรรจุข้อมูลของระดับ คอนเซ็ปชอลว์ถูกเก็บบนแผ่นแผ่นดิสก์(disk files) นอกจากนี้ระดับภายใน (internal level) บรรจุส่วนหนึ่งของหน่วยความจำฝั่งตัวโครงสร้างข้อมูลของออรากิลนั้นเรียกว่า System Global Area (SGA) และส่วนหนึ่งของ โปรแกรมพื้นหลังซึ่งคล้ายกับระดับ คอนเซ็ปชอลว์ แถวของเทเบิลใน ระดับ คอนเซ็ปชอลว์ถูกเรียกว่า ลอจิคคอลมเร็คคอร์ด และสำเนาของเขาทั้งหลายที่ระดับ อินเทอร์เน็ตถูกเรียกว่า ฟิซิคัล เร็คคอร์ด

2.2.2.6.3 เอ็กเทอร์เน็ต เลฟเว็ล(External level) ระดับภายนอกใกล้เคียงกับผู้ใช้ที่สุด มันเกี่ยวข้องกับทางในรูปแบบซึ่งผู้ใช้สามารถดูข้อมูลของเขาได้ด้วยตัวเอง ดังเช่น ผู้ใช้แตกต่างกันการเข้าถึงระดับภายนอกจึงแตกต่างกันของฐานข้อมูล ผู้ใช้สามารถเป็นผู้ใช้ปลายทาง ผู้พัฒนาฐานข้อมูล หรือ Database Administrator (DBA) ที่ทำงานบนการ Query อย่างจำเพาะเจาะจงจำนวนหนึ่ง แต่ละระดับ ภายนอก มีจำนวนมากมายแต่มีได้แค่หนึ่งเท่านั้นในระดับ คอนเซ็ปชอลว์(Conceptual Level)

2.2.2 การปรับแต่งต่าง ๆ บนระบบฐานข้อมูล (Instance Tuning) เป็นการศึกษาการปรับแต่งระบบฐานข้อมูลให้รู้ถึงวิธีการที่ถูกต้อง มีขั้นตอนตามทฤษฎีดังต่อไปนี้

การปรับแต่งแชร์พูล (Shared Pool)

การปรับแต่งบัฟเฟอร์ แคช(Buffer Cache)

การปรับแต่งไลบรารี แคช (Library Cache)

การปรับแต่งดาต้า ดิกชันนารี แคช (Data Dictionary Cache)

การปรับแต่งแต่ละลำดับชั้น(Optimization at Each Level)

2.2.3 การปรับแต่งระบบฐานข้อมูลและแสดงผล (Tuning of I/O System)

ความถี่ของการเข้าถึงบล็อกข้อมูล ออรากิลจะคาดคะเนความจำเป็นสำหรับการไปเอาบล็อก จาก ดิสก์แต่ละเวลาที่มันถูกเรียก สิ่งนี้ลด Disk I/O และเพิ่มประสิทธิภาพ โดยทั่วไปแล้วถ้า SGA ถูกจำกัดขนาดออรากิลจะใช้ แคชซึ่งอัลกอริทึม เรียกตัวที่ถูกใช้น้อยที่สุด (LRU) จัดการกับบล็อกที่อยู่ในแคชหน่วยความจำตัวที่ถูกใช้บ่อย ๆ จะยังคงอยู่ในแคช

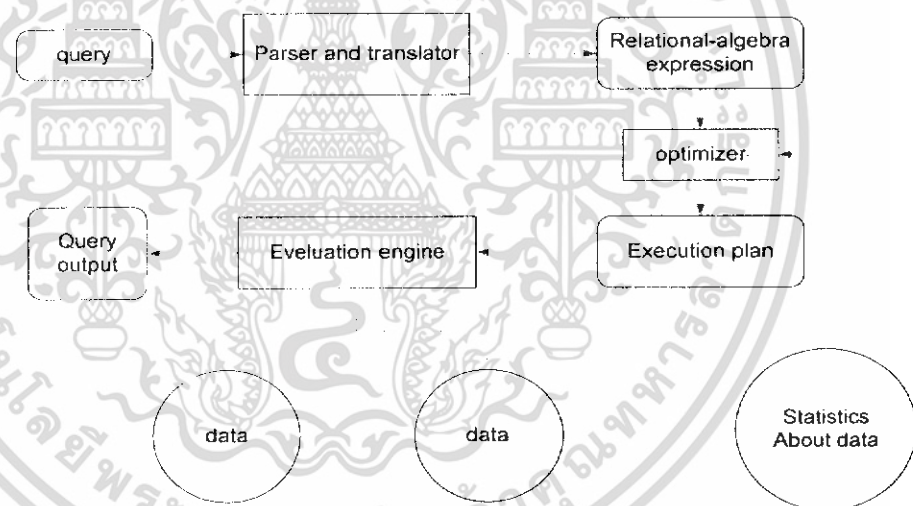
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นานเท่านั้น และตัวที่ถูกใช้น้อยจะ ต้อง โคน สวอปปี้ง(Swapping) ออกจากดิสก์เพื่อที่จะได้พื้นที่ว่างใหม่ ๆ ให้สำหรับบล็อกที่ถูกเรียก

โปรแกรมพื้นหลังมันจะรันโดยอัตโนมัติ สำหรับเหตุการณ์ส่วนมากที่เกิดขึ้นในระบบ ออราเคิลจะตัดสั้นใจจากค่าของพารามิเตอร์กระบวนการเริ่มต้น(Initialization parameter) โดยขึ้นอยู่กับค่าที่ตั้งไว้ในฐานข้อมูลที่สามารถเลือกได้สูงถึง 11 โปรแกรมพื้นหลัง

2.2.4 หลักการอ่านเขียนข้อมูลจากฐานข้อมูล

หลังจากการเปิดใช้งานฐานข้อมูลก็ย่อมจะเกิดการอ่านและเขียนข้อมูลในฐานข้อมูลอยู่เสมอ โดยการทำงานของฐานข้อมูลของ Oracle นั้นจะมีหลักการคือ ยูสเซอร์จะสามารถอ่านข้อมูลได้เฉพาะข้อมูลที่ถูกเปลี่ยนแปลงอย่างถาวรแล้วเท่านั้น (ซึ่งก็คือข้อมูลที่ Commit แล้วในฐานข้อมูล) โดยทำตามหลักการ (Read Consistency)



รูป 2-6 Step in query processing

(Database Design + Query Formulation) = Improved Performance

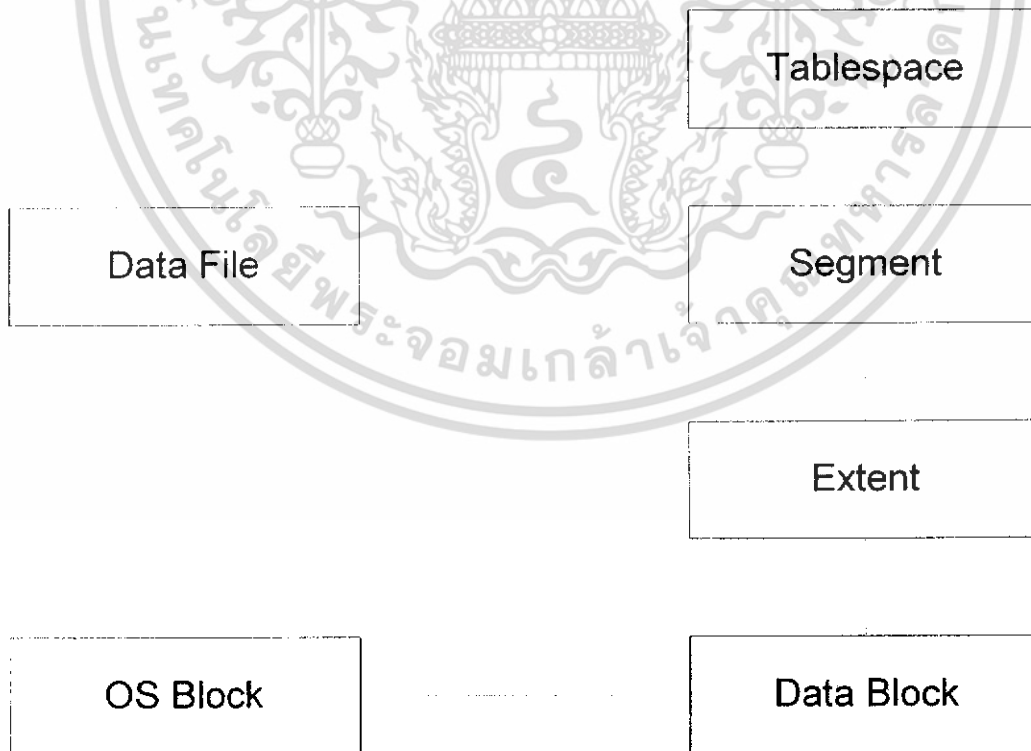
ประสิทธิภาพโดยทั่วไปแล้วจะถูกวัดโดยเวลาตอบสนอง โดยผู้ใช้ที่มีประสบการณ์ เริ่มต้นเมื่อผู้ใช้กดคีย์เข้าไปหรือ คลิกบนเมาส์และทันทีที่ตัวอักษรแรกของการตอบสนองของระบบ (เอาพุดท์จริง (desired output), เออร์ เมจเสจ (error message) ฯ) ปรากฏบนจอภาพ เวลาตอบสนองนี้ขึ้นอยู่กับสามระดับนี้ทั้งหมดซึ่งมันมีความสัมพันธ์กันตรงตาม โครงสร้างและ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สถาปัตยกรรมของฐานข้อมูลออร์เกิล ซึ่งรวมกันเรียกได้อีกอย่างหนึ่งว่า
ดาต้าเบส อินสแตนซ์ (Database Instance)

2.2.5 โครงสร้างการจัดการไฟล์และเก็บข้อมูล

โครงสร้างการจัดการไฟล์และการเก็บข้อมูลที่จะกล่าวถึงต่อไปนี้เป็นส่วนที่
เรียกว่า Oracle Database ดังที่ทราบมาแล้วว่า Oracle Database ประกอบด้วย
ไฟล์ 3 แบบ ได้แก่ Control File, Redo Log File และ data File โดย
Data File เป็นส่วนที่ใช้เก็บข้อมูลของฐานข้อมูลจริง ๆ เอาไว้ (ใช้เก็บข้อมูล
ต่าง ๆ ของ ยูสเซอร์) ซึ่งจะได้เข้าใจกับโครงสร้างของ Oracle Database ได้
เป็น 2 แบบคือ Physical Structure และ Logical Structure โดย Physical
Structure นั้นเป็นโครงสร้างจริงในระบบปฏิบัติการที่ใช้เก็บข้อมูล ได้แก่ ไฟล์
ต่าง ๆ ที่ประกอบเป็นฐานข้อมูล ส่วน Logical Structure จะเป็นโครงสร้างใน
การเก็บข้อมูลซึ่งประกอบไปด้วย Tablespace, Object, Data Block, Extent
และ Segment ขนาดของ Logical Structure สามารถเรียงลำดับจากขนาดเล็ก
ที่สุดไปหาขนาดใหญ่ที่สุดได้เป็น Data Block, Extent, Segment, และ
Tablespace ดังรูปที่ 2-7



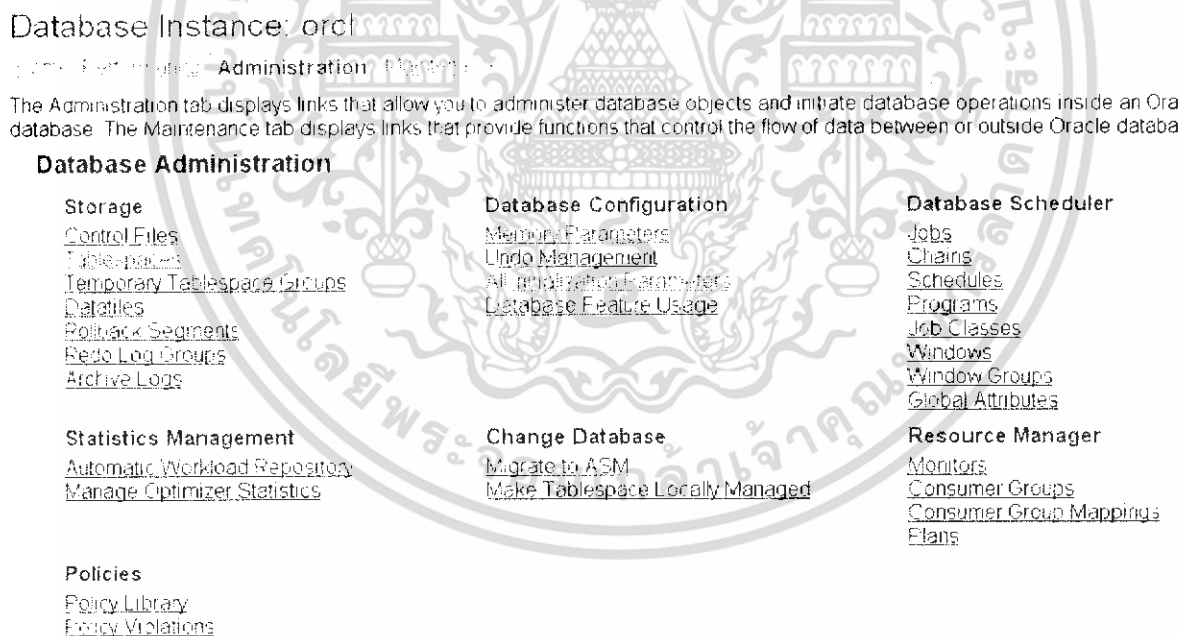
รูปที่ 2-7 แสดงความสัมพันธ์ของ Physical และ Logical Structure ของฐานข้อมูล
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป เราจะแบ่งโครงสร้างของฐานข้อมูลออกเป็น 2 ส่วนคือ Physical Structure และ Logical Structure ในส่วนของ Logical Structure โครงสร้างที่เล็กที่สุด ได้แก่ Data Block ซึ่งหลาย ๆ Data Block จะรวมกันเป็น Extent และหลาย ๆ Extent จะรวมกันเป็น Segment และหลาย ๆ Segment จะรวมกันเป็น Tablespace ในที่สุด

เมื่อมีการสร้างออบเจกต์ขึ้นใน Tablespace แต่ละออบเจกต์จะมีพื้นที่ใช้งานซึ่งก็คือ Segment นั้นเอง เมื่อจับคู่ Logical Structure กับ Physical Structure จะสามารถกล่าวได้ว่า Data Block คือ OS Block หลาย ๆ บล็อกรวมกันจนมีค่าเท่ากับค่าของ Data Block ส่วน Extent, Segment และ Tablespace ก็คือ พื้นที่การใช้งานใน Data File นั้นเอง

2.2.6 การปรับค่าพารามิเตอร์ต่าง ๆ ที่ใช้ในฐานข้อมูลของออร์เกิล

เมื่อเกิดการเชื่อมต่อกับฐานข้อมูลของออร์เกิลผ่านทาง Oracle Database Instance:orcl แล้วเข้าไปที่ TAB Administration ที่ Database Configuration คลิกเลือกที่ ALL Initialization Parameters ดังรูปที่ 2-8



รูปที่ 2- 8 ALL Initialization Parameters

หลังจากนั้นจะเข้าไปสู่ Initialization Parameter สำหรับยูสเซอร์ที่เป็น SYSDBA เท่านั้นที่มีสิทธิสูงสุดในการปรับแต่งพารามิเตอร์ได้ทั้งหมด จะมี ช่องว่าง และ ลิสบ็อกให้ใส่ ค่าที่ต้องการปรับแต่งฐานข้อมูลได้ดังรูป หรือจะปรับทาง SQL*Plus โดยใช้คำสั่ง ALTER SYSTEM ดังรูปที่2-10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

db_name	str	orcl	String	✓	✓	Database Identification
ol_recovery_file_dest	str	E:\oracle\product\10.2.0\flash_f	String	✓	✓	Backup and Recovery
ol_recovery_file_dest_size	int	2G	Big Integer	✓	✓	Backup and Recovery
open_cursors	int	300	Integer	✓	✓	Cursors and Library Cache
pga_aggregate_target	int	700M	Big Integer	✓	✓	Memory
processes	int	150	Integer	✓	✓	Processes and Session
remote_login_passwordfile	str	EXCLUSIVE	String	✓	✓	Security and Auditing
sga_target	int	160M	Big Integer	✓	✓	Memory
undo_management	str	AUTO	String	✓	✓	Automatic Undo Management
undo_tablespace	str	UNDOTBS1	String	✓	✓	Automatic Undo Management
cluster_database	bool	FALSE	Boolean	✓		Cluster Database
db_create_file_dest	str		String	✓	✓	File Configuration
db_create_online_log_dest_1	str		String	✓	✓	File Configuration
db_create_online_log_dest_2	str		String	✓	✓	File Configuration
instance_number	int	1	Integer			Cluster

รูปที่ 2-9 ช่องว่าง และ ลิสต์บอกที่ใช้ค่าที่ต้องการจะปรับค่าพารามิเตอร์

```

Oracle SQL*Plus
File Edit Search Options Help

SQL*Plus: Release 10.2.0.1.0 - Production on อ. น.ค. 6 18:19:12 2007
Copyright (c) 1982, 2005, Oracle. All rights reserved.

เซ็่มพ์อ์ลัษ:
Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production
With the Partitioning, OLAP and Data Mining options

SQL> alter system;

```

รูปที่ 2-10 ปรับทง SQL*Plusโดยใช้คำสั่ง ALTER SYSTEM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การปรับแต่งประสิทธิภาพและจัดการฐานข้อมูล

3.1 Basic Index Principles

ส่วนนี้จะพิจารณาที่ basic index theory เพื่อให้ผู้เริ่มต้นมีพื้นฐานและมองเห็นถึงประสิทธิภาพของการใช้งานอินเด็กซ์(index)ของ Oracle Oracle มีอยู่หลาย indexing option การเลือกออกแบบให้เหมาะสมกับสถานการณ์สามารถชี้ขาดได้ถึงประสิทธิภาพของแอปพลิเคชัน(application) เนื้อหาส่วนนี้จะพิจารณาแต่ละออกแบบว่ามีข้อดีและข้อจำกัดอะไรบ้าง

3.1.1 Basic Index Concepts

เมื่อเข้าถึงข้อมูลในเทเบิล(table) Oracle มี 2 ทางเลือกคือ

- 1.เข้าถึงทุกๆ row ในเทเบิล(full table scan)
- 2.เข้าถึงแต่ละ row โดยใช้ ROWID(index-based access)

เมื่อเข้าถึงน้อยกว่า 5% ของบล็อก(block)ทั้งหมดอินเด็กซ์อาจถูกใช้ ถ้าข้อมูลที่ต้องการเข้าถึงกระจายอยู่ในหลายๆบล็อกประสิทธิภาพจาก full table scan อาจดีกว่าการเข้าถึงโดยใช้อินเด็กซ์ optimizer จะเป็นคนตัดสินใจว่าจะใช้เส้นทางใดโดยจะเลือกเส้นทางที่เหมาะสมที่สุด

ในบางออกแบบใหม่ที่เริ่มใช้ตั้งแต่ Oracle 8 เช่น partitioning, parallel DML และ I/O ที่ใหญ่ขึ้นที่ใช้ใน DB_FILE_MULTIBLOCK_READ_COUNT ทำให้จุดสมดุระหว่าง full table และ index เปลี่ยนไป

โดยทั่วไปอินเด็กซ์จะช่วยเพิ่มประสิทธิภาพสำหรับคำสั่ง SELECT, UPDATE, DELETE(เมื่อเข้าถึงข้อมูลจำนวนไม่มาก) และลดประสิทธิภาพของคำสั่ง INSERT

คำสั่ง INSERT จะมีการอินเสิร์ต(insert)ข้อมูลในเทเบิลและในอินเด็กซ์ โดยทั่วไปแต่ละอินเด็กซ์ที่อินเด็กซ์บนเทเบิลจะทำให้คำสั่ง INSERT ช้าลงประมาณ 3 เท่า ปัจจุบันอินเด็กซ์ที่มีส่วนประกอบ 2 ส่วนจะช้ากว่าอินเด็กซ์ที่มีส่วนประกอบส่วนเดียวไม่มาก

คำสั่ง UPDATE ถ้าทำการอัปเดต(update)บนคอลัมน์(column)ที่ถูกอินเด็กซ์ จะทำให้เกิดการอินเสิร์ตและ delete ในอินเด็กซ์ การอัปเดตบนคอลัมน์ที่ถูกอินเด็กซ์จะทำให้คำสั่ง UPDATE ช้าลงประมาณ 3 เท่าขึ้นอยู่กับสถาปัตยกรรมและการเก็บข้อมูล

อย่างไรก็ตามคำสั่ง INSERT จะทำงานช้ากว่าคำสั่ง UPDATE เพราะคำสั่ง INSERT มีการอินเสิร์ตข้อมูลในเทเบิลและทุกๆอินเด็กซ์ของเทเบิลนั้น ในขณะที่คำสั่ง UPDATE มีการอินเสิร์ตที่อินเด็กซ์เพียงตัวเดียว(เมื่ออัปเดตบนคอลัมน์ที่ถูกอินเด็กซ์)

โอเวอร์เฮด(overhead) ของการอินเสิร์ตข้อมูลจะมาก ส่วนจะมากแค่ไหนจะขึ้นอยู่กับโครงสร้างการจัดเก็บของข้อมูลนั้น

เอกสารนี้เป็นเอกสารทบทวนไวสำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 Concatenated Indexes

เมื่ออินเด็กซ์ทำการอินเด็กซ์กับหลายคอลัมน์จะเรียกอินเด็กซ์นั้นว่า concatenated index หรือ composit index concatenated index จะถูกใช้เมื่อ leading column ของอินเด็กซ์ถูกใช้ใน WHERE clause ถ้าใน WHERE clause ไม่ใช้ leading column ของอินเด็กซ์แต่ยังมีการใช้ส่วนอื่นของอินเด็กซ์อยู่ Oracle จะทำการ fast full scan of the index แทนการ full table scan

VENDOR_PARTS		
VEND ID	PART NO	UNIT COST
1012	10-440	.25
1012	10-441	.30
1012	457	4.95
1010	10-440	.27
1010	457	5.10
1220	08-300	1.33
1012	08-300	1.19
1292	457	5.28

Concatenated Index
(index with multiple columns)

รูปที่ 3-1 Concatenated Indexes

3.1.3 The Oracle ROWID

ROWID คือเส้นทางที่ชี้ตรงไปยังที่อยู่ของแต่ละ row เมื่อมีการ reorganize เทเบิลจะทำให้ ROWID เปลี่ยนไปซึ่งทำให้ค่า ROWID ที่เคยเก็บไว้ในแอปพลิเคชันจะไม่สามารถใช้งานได้ ดังนั้นจึงควรระมัดระวังเมื่อเขียนโปรแกรมทำงานกับ ROWID

ROWID ใน index-organized table (IOT) ไม่ใช่ physical ROWID นั้นหมายความว่า ROWID นี้จะชี้ไปยัง row ส่วนที่เหลือของเทเบิลที่ไม่ได้อยู่ใน IOT จะพุดถึง IOT อีกครั้งในส่วนถัดไป

ข้างล่างนี้คือตัวอย่าง ROWID ที่อยู่ในฐานข้อมูลซึ่งได้ทำการควอรี(query) นำออกมาดู
SELECT ROWID, last_name FROM employees WHERE department_id = 20;

```
ROWID          LAST_NAME
-----
```

```
AAAAaoAATAAABrXAAA BORTINS
```

```
AAAAaoAATAAABrXAAE RUGGLES
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AAAAaoAATAAABrXAAG CIEN

AAAAaoAATAAABrXAAN BLUMBERG

OOOOOFFFFBBBBB}BRRR

OOOOOO data object number

FFF datafile number

BBBBBB data block

RRR row

3.1.4 Suppressing Indexes

หลายๆกับดักใน SQL ทำให้อินเด็กซ์ไม่ถูกใช้งาน ส่วนต่อไปนี้จะพิจารณาปัญหาที่พบบ่อยซึ่งมักจะเกิดจากการขาดประสิทธิภาพของ โปรแกรมเมอร์

การใช้ NOT EQUAL Operator (<>, !=)

อินเด็กซ์จะใช้ในการหาข้อมูลที่มีอยู่ในเทเบิลเท่านั้น เมื่อ NOT EQUAL ถูกใช้กับคอลัมน์ที่ถูกอินเด็กซ์ใน WHERE clause จะทำให้อินเด็กซ์ที่ทำการอินเด็กซ์อยู่บนคอลัมน์นั้นไม่ถูกใช้งาน Oracle จะทำ full table scan เพราะทุก record ต้องถูก retrieve

การเปลี่ยน <> ให้เป็น < or > อาจทำให้ optimizer ใช้งานอินเด็กซ์ เพื่อให้ได้ประสิทธิภาพที่ดีควรทำให้แน่ใจว่าผลลัพธ์ที่ได้จะทำให้เกิดการ retrieve ยังคงน้อยกว่ามีค่าน้อยกว่า 5 % ของเทเบิล

Using IS NULL or IS NOT NULL

การใช้ IS NULL หรือ IS NOT NULL กับคอลัมน์ที่ถูกอินเด็กซ์ จะทำให้อินเด็กซ์ที่ทำการอินเด็กซ์อยู่บนคอลัมน์นั้นไม่ถูกใช้งาน เพราะค่าที่เป็น NULL จะไม่ถูกบันทึกไว้ในอินเด็กซ์ ยกเว้นอินเด็กซ์ที่เป็น bitmap index

การกำหนดให้คอลัมน์ไม่ใช่ค่า NULL ทำได้โดยการกำหนดให้คอลัมน์นั้นเป็น NOT NULL ในขณะที่ทำการ create table หรือ alter table

คำสั่ง ALTER TABLE ถ้าข้อมูลในคอลัมน์นั้นมีข้อมูลครบอยู่แล้วเราสามารถใส่ NOT NULL กับคอลัมน์นั้นได้เลย แต่ถ้าในคอลัมน์นั้นข้อมูลไม่ครบเราต้องใช้ DEFAULT cluse ของคำสั่ง ALTER TABLE ช่วยในการกำหนดค่า default

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Using Functions

การใช้ฟังก์ชัน(function)กับคอลัมน์ที่ถูกอินเด็กซ์ ใน WHERE clause จะทำให้อินเด็กซ์ที่ทำการอินเด็กซ์อยู่บนคอลัมน์นั้นไม่ถูกใช้งาน เพราะฟังก์ชันจะทำการแปลงค่าของข้อมูลทั้งหมดที่อยู่ในคอลัมน์นั้น row ทั้งหมดของเทเบิลจึงต้องถูกเข้าถึง ดังนั้นอินเด็กซ์จึงไม่ถูกใช้งาน เราควรหลีกเลี่ยงการใช้ฟังก์ชันกับคอลัมน์โดยตรงเพื่อประสิทธิภาพที่ดี

กำหนดให้ hiredate เป็นคอลัมน์

$\text{TRUNC}(\text{hiredate}) \geq '10\text{-MAY-81}'$: มีการใช้ฟังก์ชันกับคอลัมน์

hiredate

$\text{hiredate} < (\text{TO_DATE}('10\text{-MAY-81}') + 0.9999)$: ใช้ฟังก์ชันกับค่าที่ใช้เปรียบเทียบกับคอลัมน์

Comparing Mismatched Data Types

การเปรียบเทียบข้อมูลที่มีชนิดต่างกัน Oracle จะทำการแปลงข้อมูลนั้นให้มีชนิดเหมือนกันก่อนทำการเปรียบเทียบ เมื่อมีการแปลงข้อมูลในคอลัมน์ก็จะเหมือนกับการทำฟังก์ชันกับคอลัมน์นั้น อินเด็กซ์ที่ทำการอินเด็กซ์บนคอลัมน์นั้นจะไม่ถูกใช้งาน ดังนั้นการเปรียบเทียบควรจะใช้ชนิดข้อมูลที่เหมือนกันในการเปรียบเทียบ

กำหนดให้ account_number มีชนิดเป็น VARCHAR2 account_number = 990354 จะถูกแปลงเป็น to_number(account_number) = 99354 ซึ่งทำให้อินเด็กซ์ไม่ถูกใช้งาน การเปรียบเทียบที่ถูกต้องคือ account_number = '990354'

3.1.5 Selectivity

การตัดสินใจว่าจะใช้งานอินเด็กซ์หรือไม่ขึ้นอยู่กับ query และข้อมูล ค่าหนึ่งที่ช่วยในการตัดสินใจคือจำนวนของ unique key ในอินเด็กซ์ เราจะต้องทำการ analyze table หรือ analyze index ก่อนถึงจะได้ค่านี้ ค่านี้จะถูกเก็บไว้ในคอลัมน์ distinct_keys ของ USER_INDEXES view ถ้าค่า selectivity ยิ่งมากจำนวน row ที่รีเทอร์น(return)ก็จะน้อยตามไปด้วย

การเพิ่มค่า selectivity ให้กับอินเด็กซ์ทำได้โดยการทำอินเด็กซ์นั้นให้เป็น concatenate index หลักในการทำ concatenate index คือ เมื่อทำการเพิ่มคอลัมน์ที่ถูกอินเด็กซ์ การเพิ่มนั้นจะต้องทำให้ค่า selectivity ของอินเด็กซ์เพิ่มขึ้น ถ้าค่า selectivity ไม่เพิ่มขึ้นจากการเพิ่ม

คอลัมน์จะทำให้อินเด็กซ์มีประสิทธิภาพลดลง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สงวนลิขสิทธิ์เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.6 The Clustering Factor

ค่า clustering factor จะถูกเก็บในอยู่ในคอลัมน์ clustering_factor ของ USER_INDEXES view ถ้าค่านี้เข้าใกล้จำนวน leaf block ของอินเด็กซ์แสดงว่าข้อมูลถูกเรียงอย่างดีในเทเบิล(เรียงตามอินเด็กซ์) แต่ถ้าค่านี้เข้าใกล้จำนวน row ของเทเบิลแสดงว่าข้อมูลนั้นถูกเรียงไม่ดี จำนวนที่มากขึ้นของ Clustering Factor ทำให้จำนวนของบล็อกที่ต้องการในการ query มีมากขึ้น

ข้อมูลที่มีการจัดเรียงดีจะเหมาะกับคำสั่งประเภท range-scan-type operation ซึ่งสามารถทำได้โดยการทำคลัสเตอร์(cluster)ของข้อมูลภายในเทเบิล

3.1.7 The Binary Height

ค่า binary height ของอินเด็กซ์แสดงให้เห็นว่าจะต้องใช้ I/O เท่าใดในการรีเทิร์น ROWID ออกมาจากอินเด็กซ์ เนื่องจากแต่ละบล็อกในอินเด็กซ์ไม่ได้ถูกอ่านตามลำดับดังนั้นแต่ละบล็อกจะต้องใช้ 1 I/O ในการเข้าถึง

ค่า binary height ของอินเด็กซ์จะถูกเก็บในคอลัมน์ b-level ของ DBA_INDEXES view เราจะต้องทำการ analyze table หรือ analyze index ก่อนถึงจะได้ค่านี้

binary height จะเพิ่มขึ้นตามขนาดของเทเบิลและ range ของคอลัมน์ที่ถูกอินเด็กซ์อย่างประหยัด การ delete ข้อมูลในเทเบิลทำให้เกิดการ delete ในอินเด็กซ์ด้วย การ delete จำนวนมากในอินเด็กซ์ทำให้ค่า binary height สูงกว่าที่ควรจะเป็น เช่นเดียวกับการอัปเดตบนคอลัมน์ที่ถูกอินเด็กซ์ก็ทำให้เกิดการ delete และอินเด็กซ์ในอินเด็กซ์ เมื่อมีการ delete ภายในอินเด็กซ์ 20-30% การ rebuilt index จะช่วยทำให้ค่า binary height ลดลง ถ้าขนาดของ database block size มีขนาดใหญ่จะทำให้ค่า binary height ของอินเด็กซ์น้อย binary height ยิ่งเพิ่มขึ้นก็จะยิ่งทำให้ประสิทธิภาพของ DML ลดลง

3.1.8 Using Histograms

histogram ช่วย cost-base optimizer ในการวางแผนการทำงานว่าเมื่อไหร่ควรจะใช้อินเด็กซ์ ในกรณีที่ข้อมูลในเทเบิลมีการกระจุกตัวของข้อมูลในช่วงใดช่วงหนึ่งในจำนวนที่มาก เราสามารถสร้าง histogram ได้กับทุกคอลัมน์ไม่ว่าคอลัมน์นั้นจะถูกอินเด็กซ์หรือไม่

เมื่อทำการสร้าง histogram สิ่งที่ต้องกำหนดคือ size โดยที่ size คือจำนวน bucket ใน histogram แต่ละ bucket จะเก็บค่าของคอลัมน์และจำนวน row โดย default จะทำการสร้าง histogram ที่มีขนาด 75 bucket เราสามารถกำหนดจำนวน bucket ได้ตั้งแต่ 1- 254 histogram ของ Oracle เป็นชนิด height balance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.9 Fast Full Scans

fast full scan เป็นออปรแกรมที่ Oracle ทำ full index scan operation โดยจะทำการอ่าน leaf block ทั้งหมดของ b-tree index ตามลำดับ พารามิเตอร์

DB_FILE_MULTIBLOCK_READ_COUNT จะเป็นตัวควบคุมจำนวนบล็อกที่สามารถอ่านขึ้นมาได้พร้อมกัน ดังนั้นการทำ full index scan จะมีการอ่าน leaf block ขึ้นมาพร้อมกันทีละหลายๆบล็อก

fast full scan จะถูกใช้เมื่อทุกคอลัมน์ใน query อยู่ในอินเด็กซ์และใน WHERE clause ไม่มี leading column ของ concatenated index อยู่

ถ้า index มีขนาดเล็กเมื่อเทียบกับเทเบิล fast full scan จะมีประสิทธิภาพดีกว่าเมื่อเทียบกับ full table scan

fast full scan มักจะถูกทำในขณะที่เกิดการ join ในกรณีที่ใช้เฉพาะคอลัมน์ที่อยู่ในอินเด็กซ์

3.1.10 Skip-Scan

index skip-scan feature ทำให้ optimizer ใช้งาน concatenated index ในกรณีที่ leading column ไม่ถูกใช้ใน WHERE clause skip-scan จะเร็วกว่า fast full scan ในกรณีที่ต้องการอ่านข้อมูลในจำนวนที่ไม่มาก

เราสามารถโน้มน้าวให้ optimizer ทำ skip-scan ได้โดยการบอกไป optimizer จะเป็นคนตัดสินใจว่าจะทำตามที่เราแนะนำเสนอหรือไม่

3.1.11 B-tree Indexes

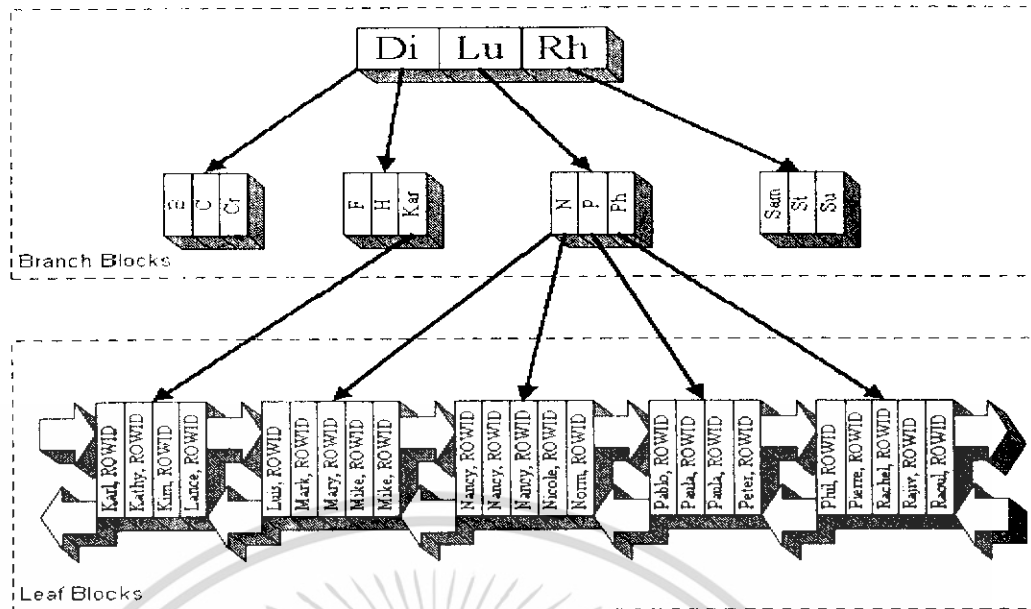
b-tree index เป็นอินเด็กซ์ที่มีโครงสร้างแบบต้นไม้ ซึ่งสามารถอินเด็กซ์ได้ทีละคอลัมน์หรือทีละหลายคอลัมน์(มากที่สุด 32 คอลัมน์) อินเด็กซ์ชนิดนี้เป็น default ของ Oracle เมื่อมีการสร้างอินเด็กซ์

ที่ branch block แต่ละ row จะเก็บ block ID ของบล็อกใน level ถัดไป

ที่ leaf block จะเก็บ index value, ROWID และ data block address(DBA) ของ next leaf block และ previous leaf block

b-tree index จะเก็บ ROWID ของทุกๆ row ที่มีค่าอยู่ในคอลัมน์ที่ถูกอินเด็กซ์ ดังนั้น Oracle จะไม่อินเด็กซ์ไปยังค่า NULL ดังรูปที่ 3-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-2 B-tree Indexes

3.1.12 Bitmap Indexes

เมื่อทำการสร้าง bitmap index Oracle จะทำการสร้าง bitmap สำหรับแต่ละ distinct key และเก็บ ROWID ที่สัมพันธ์กันคีย์(key)นั้นไว้ bitmap เหมาะกับคอลัมน์ที่มีค่า cardinality ที่ไม่มาก การเข้าถึงข้อมูลในกรณีนี้โดยใช้ bitmap index จะเร็วกว่า b-tree index เพราะในกรณีนี้ bitmap index จะมีขนาดเล็กกว่า b-tree index bitmap index สามารถอินเด็กซ์ได้มากที่สุด 30 คอลัมน์แต่โดยทั่วไปมักจะใช้อินเด็กซ์ทีละไม่กี่คอลัมน์

ประโยชน์หลักของ bitmap index คือทำให้ Oracle สามารถรวมเซต(set)ของผลลัพธ์ที่ได้จากแต่ละ bitmap index เพื่อตัดผลลัพธ์ที่ไม่ต้องการได้อย่างรวดเร็ว ในกรณีที่เทเบิลมี bitmap index อยู่หลายอัน

การอินเด็กซ์ของ bitmap index จะเร็วกว่า b-tree index 3-4 เท่าในกรณีที่คอลัมน์มี cardinality ไม่มาก และ b-tree index จะเร็วกว่า bitmap index ในกรณีที่มีการอินเด็กซ์ค่าใหม่ที่ไม่ซ้ำกับของเดิมมากกว่า 70%

bitmap index เหมาะกับงานที่ใช้ดูข้อมูลเป็นส่วนใหญ่ เช่น decision support system(DDS) และ data warehouse bitmap index ไม่เหมาะกับงานที่มีการเปลี่ยนแปลงข้อมูลบ่อยๆ เช่น online transaction processing(OLTP) เพราะ bitmap index อาจทำให้เกิด deadlock คำสั่ง ALTER TABLE ถ้าทำบนคอลัมน์ที่ถูกอินเด็กซ์โดย bitmap index จะมีผลทำให้ bitmap index นั้นใช้งานไม่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CUSTOMER #	MARITAL_STATUS	REGION	GENDER	INCOME_LEVEL
101	single	east	male	bracket_1
102	married	central	female	bracket_4
103	married	west	female	bracket_2
104	divorced	west	male	bracket_4
105	single	central	female	bracket_2
106	married	central	female	bracket_3

รูปที่ 3-3 Bitmap Indexes



รูปที่ 3-4 Bitmap Indexes

status = 'married'	region = 'central'	region = 'west'						
0	0	0		0	0	0		0
1	1	0		1	1	1		1
1	0	1		1	1	1		1
0	0	1	AND		=	0	AND	0
0	1	0		0	0	1		0
1	1	0		1	1	1		1

รูปที่ 3-5 Bitmap Indexes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.13 Hash Indexes

เมื่อ hash cluster ถูกสร้าง cluster key จะถูกกำหนดให้ คีย์นี้จะบอกให้ Oracle ทราบว่าจะเก็บเทเบิลอย่างไรใน hash cluster เมื่อทำการเก็บข้อมูล row ที่สัมพันธ์กับ cluster key เหมือนกันจะถูกเก็บไว้ใน database block เดียวกัน การเข้าถึงข้อมูลโดยใช้ hash index Oracle จะใช้ hash function และ I/O ในการเข้าถึงข้อมูล

hash index ทำให้ Oracle เข้าถึงข้อมูลได้เร็วแต่มันก็มีข้อเสีย จำนวนdistinct value จะต้องถูกกำหนดในตอน create ถ้าค่าที่กำหนดน้อยเกินไปจะทำให้เกิด collision ซึ่งทำให้ข้อมูลที่ควรจะถูกเก็บไว้ใน cluster เดียวกันดันเก็บไว้ที่อื่นและต้องใช้ I/O เพิ่มขึ้นในการเข้าถึงข้อมูลนั้น

hash index มีจุดประสงค์เพื่อประหยัดพื้นที่ในการจัดเก็บข้อมูล ถ้าไม่รู้จำนวนที่แน่นอนของ distinct อาจทำให้เกิดการสิ้นเปลืองพื้นที่ hash index จะไม่เหมาะสมถ้าหากไม่สามารถหาพื้นที่เพิ่มได้ในกรณีที่ hash index ขยายตัว

ถ้าแอปพลิเคชันทำการ full table scan บ่อยๆกับเทเบิลที่ถูกทำ hash cluster ในกรณีนี้ hash cluster อาจไม่เหมาะสมเพราะใน hash cluster อาจจะมีช่องว่างที่เอาไว้รองรับการขยายตัว ช่องว่างเหล่านี้ทำให้ประสิทธิภาพของการทำ full table scan ลดลงอย่างมาก

ก่อนจะทำ hash index ต้องแน่ใจว่าข้อมูลนั้นเหมาะกับการทำ hash cluster โดยทั่วไป hashing จะทำงานได้ดีที่สุดกับข้อมูลที่เป็นลำดับ

3.1.14 Index-Organized Table

IOT เป็นเทเบิลที่ถูกแปลงจากเทเบิลมาเก็บไว้ใน b-tree index โดยเรียงข้อมูลตาม primary key ของเทเบิล ส่วนของเทเบิลที่เหลือจะถูกเก็บไว้ใน unorganized heap ข้อมูลที่เก็บไว้ใน heap จะถูกเชื่อมต่อกับ IOT โดยผ่านทาง ROWID

ทุกคอลัมน์ในคำสั่ง CREATE TABLE จะถูกเก็บใน IOT IOT ทำให้การเข้าถึงข้อมูลโดยใช้คีย์เร็วขึ้นทั้งแบบ exact match และ range search คำสั่ง UPDATE และ DELETE อาจมีประสิทธิภาพเพิ่มขึ้น

การเข้าถึงข้อมูลที่อยู่ใน heap จะใช้เวลานาน ดังนั้นข้อมูลที่จะเก็บอยู่ใน heap ควรเป็นข้อมูลที่ไม่ถูกใช้บ่อยๆ

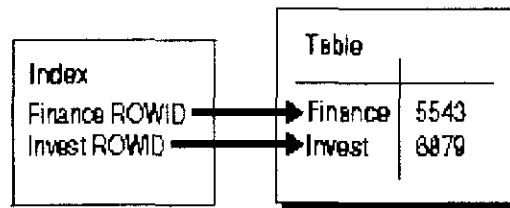
IOT สามารถถูกกระทำด้วย DML และ DDL เหมือนเทเบิลทั่วไป

IOT สามารถถูกทำ partition ได้

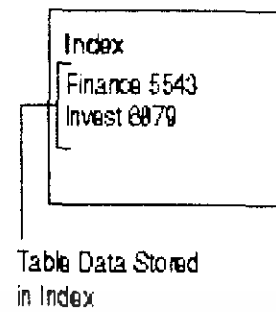
IOT สามารถถูกอินเด็กซ์โดย bimap index และ b-tree index ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

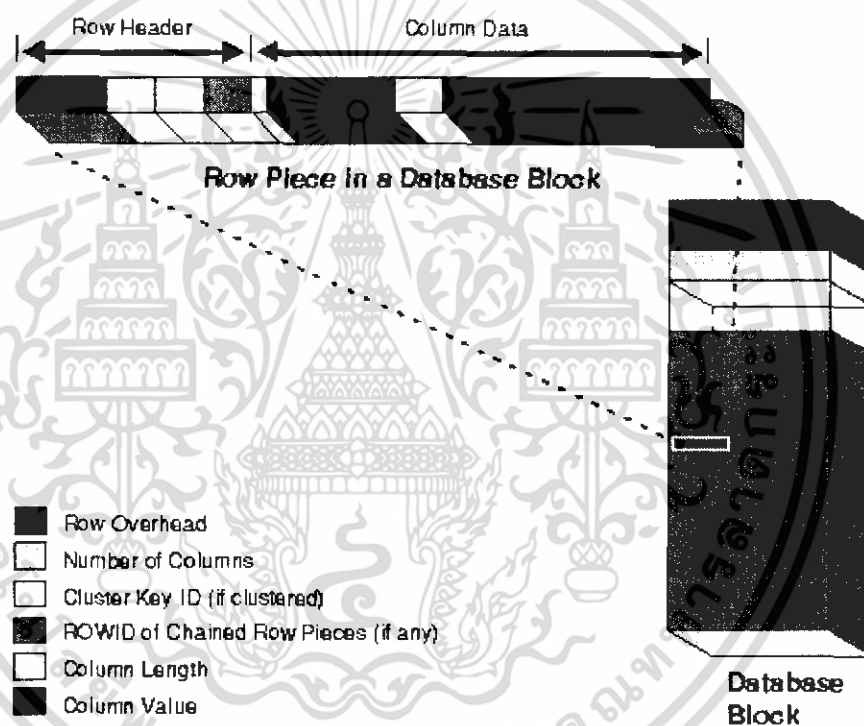
Regular Table and Index



Index-Organized Table



รูปที่ 3-6 Index-Organized Table



รูปที่ 3-7 Index-Organized Table

3.1.15 Function-Based Indexes

เมื่อ query ใช้ฟังก์ชันกับคอลัมน์จะทำให้อินเด็กซ์ที่อินเด็กซ์บนคอลัมน์นั้นไม่ถูกใช้งาน function-based index ทำให้ function-based query สามารถใช้งานอินเด็กซ์ได้ การสร้างอินเด็กซ์นี้ทำได้โดยการกำหนดค่าในขั้นตอนการ create

Create index EMP\$UPPER_JOB on

Emp (UPPER(job));

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function-based index จะถูกใช้โดย optimizer การทำให้ optimizer ทำงานในโหมด (mode) นี้ได้จะต้องเซตพารามิเตอร์(parameter) QUERY_REWRITE_ENABLED ให้เป็น TRUE (defaultจะเป็นค่า FALSE)

ก่อนจะใช้ต้องพิจารณาถึง

-สามารถควบคุมฟังก์ชันที่ใช้งานกับคอลัมน์นั้นได้หรือไม่

-มีพื้นที่ในการจัดเก็บเพียงพอหรือไม่

-เมื่อต้องการ drop เทเบิลจะต้องใช้เวลาในการ drop อินเด็กซ์มากขึ้น เวลาที่ใช้มากขึ้น

ในการ drop เทเบิลสามารถรับได้หรือไม่

3.1.16 Partitioned Index

partitioned index คือ อินเด็กซ์ที่ถูกแบ่งเป็นชั้นเล็กๆหลายๆชั้น แต่ละชั้นอาจถูก เก็บไว้ใน disk ที่แตกต่างกัน ซึ่งทำให้ I/O ที่ใช้ลดลง b-tree index และ bitmap index สามารถถูกทำ พาทิชัน (partition), เทเบิลถูกทำ พาทิชัน แต่ อินเด็กซ์ถูกทำพาทิชัน หรือทั้งเทเบิลและอินเด็กซ์ ถูกทำพาทิชัน การทำพาทิชันจะช่วยเพิ่มประสิทธิภาพและลดความสามารถในการบำรุงรักษา

partitioned index มี 2 ชนิดคือ local และ global แต่ละชนิดมี 2 subset คือ prefixed และ nonprefixed เทเบิล สามารถมีอินเด็กซ์ที่มีชนิดที่แตกต่างกัน อินเด็กซ์ที่คอลัมน์ของเทเบิล เหตุผลหลักของการทำ พาทิชันอินเด็กซ์ (partitioned index) คือ ทำให้อินเด็กซ์ที่ต้องการอ่านมีขนาดเล็กลง และความสามารถในการเก็บแต่ละพาทิชันไว้ในแต่ละเทเบิลสเปซ (Tablespace) ช่วยทำให้ reliability และ availability ดีขึ้น

ออรากเคิล มีความสามารถทำ parallel query และ parallel DML เมื่อใช้พาทิชันเทเบิล (partitioned table) และ พาทิชันอินเด็กซ์ การทำงานแบบนี้สามารถใช้ Multiple process ช่วยในการประมวลผลของคำสั่งได้ จำนวนมากที่สุดของ parallelism ขึ้นอยู่กับจำนวนของพาทิชัน

3.1.17 Local Indexes (ถูกใช้ในกรณีปกติ, commonly used)

Local index ถูก partitioned โดยใช้ partition key และขอบเขตข้อมูลที่เหมือนกับของ เทเบิล แต่ละพาทิชันของ Local indexes จะเก็บคีย์และ ROWID ที่ตรงกับ table partition ของมันเท่านั้น local indexes สามารถเป็น b-tree indexes หรือ bit-map indexes ถ้ามันเป็น b-tree indexes มันสามารถเป็น unique หรือ non-unique

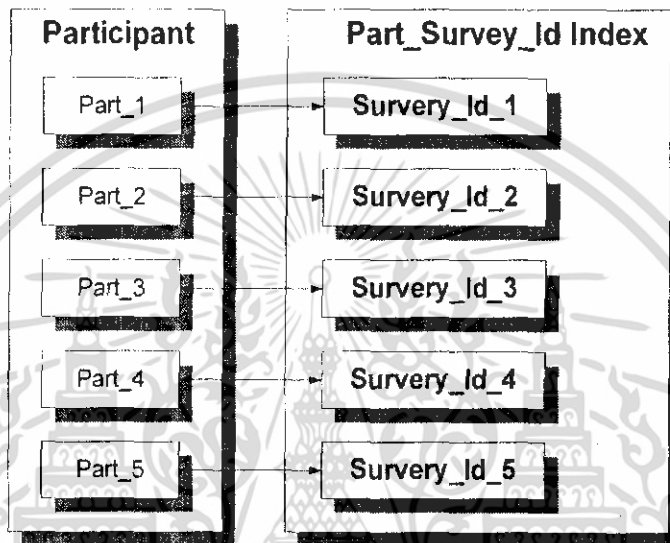
อินเด็กซ์ ชนิดนี้สนับสนุน partition independence หมายความว่าแต่ละพาทิชัน สามารถถูก add, truncate, drop, split, taken, offline โดยปราศจากการ drop หรือ rebuilt index ดังนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ละ local index partition สามารถ rebuilt โดยที่พาทิชันอื่นไม่ต้องถูก rebuilt ได้ ออราเคิลจะ maintain local indexes ให้โดยอัตโนมัติ

3.1.18 Prefixed

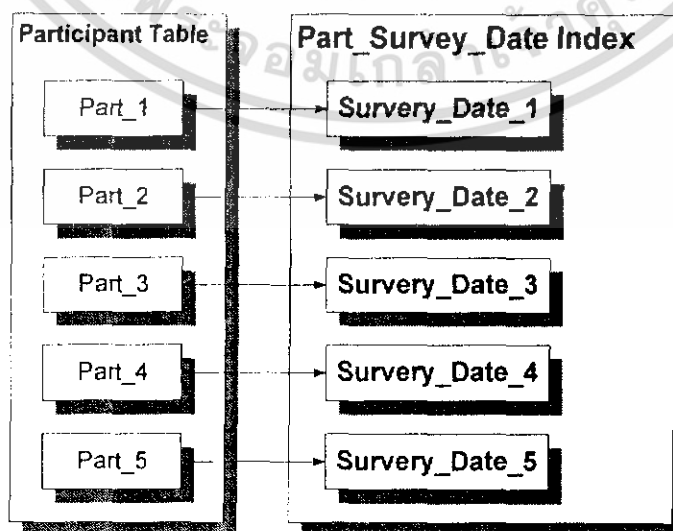
Prefixed index จะเก็บ key จาก partitioning key ในตำแหน่ง leading column ของอินเด็กซ์ local prefixed index ทำให้ออราเคิล สามารถตัด partition ที่ไม่ต้องการ ได้อย่างรวดเร็ว



รูปที่ 3-8 prefixed partitioned index

3.1.19 Non prefixed

non prefixed indexes จะไม่มี leading column ของ partition key ใน leading column ของ index สำหรับ nonprefixed index ที่เป็น unique มันจะต้องเก็บ subset ของ partition key



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3-9 Non prefixed partitioned index ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.20 Global Indexes

Global partitioned indexes จะเก็บ key จากหลาย ๆ table partitions ใน single index partition partitioning key ของ global partitioned index จะแตกต่างกันหรือถูกทำด้วยช่องทางของข้อมูลที่แตกต่างกัน ผู้สร้างจะต้องรับผิดชอบในการกำหนด rang และ value ของ partitioning key เอง global indexes สามารถเป็น b-tree indexes ได้เช่นเดียว global index ไม่ถูก maintain โดยออรากิล ดังนั้น ถ้า partition ถูกทำ add, truncate, drop, split อินเด็กซ์จะต้องถูก rebuilt ดังนั้นถ้าต้องการใช้ equi-partitioned เราควรจะสร้าง อินเด็กซ์ ให้เป็น local index ออรากิล ไม่สนับสนุน nonprefixed global indexes

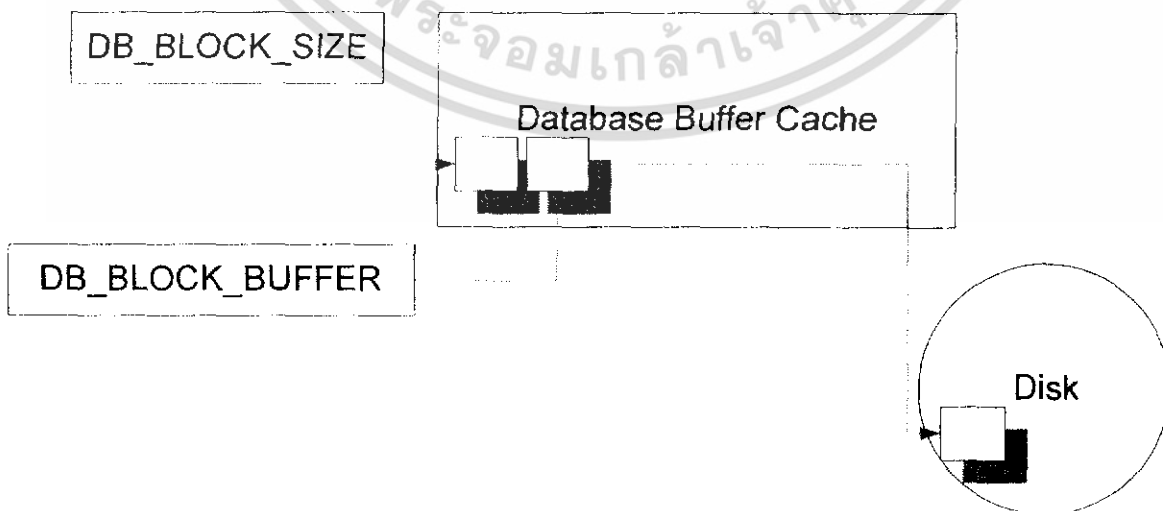
3.2 การปรับแต่งฐานข้อมูลกับ Initialization parameters

พารามิเตอร์ต่าง ๆ นี้เป็นส่วนหนึ่งของพารามิเตอร์ไฟล์อยู่ใน (INIT.ORA Files)

3.2.1 เพิ่มประสิทธิภาพโดยการปรับตั้งค่า DB_CACHE_SIZE

เมื่อผู้ใช้ ร้องขอข้อมูล ข้อมูลจะถูกนำมาเก็บไว้ในหน่วยความจำ เมื่อข้อมูลเต็มก็จะนำข้อมูลที่ถูกใช้น้อยที่สุดออกไป (LRU) และเมื่อต้องการใช้ข้อมูลนั้นอีกครั้งก็ต้องนำข้อมูลเข้ามาใหม่ ซึ่งต้องใช้ I/O การนำข้อมูลกลับมาจากหน่วยความจำเร็วกว่านำกลับมาจากดิสก์ 10,000 เท่า การทำดิสก์แคชซึ่ง (disk caching 'memory on disk') จะทำให้ดีกว่าการอ่านข้อมูลจากหน่วยความจำที่ยังไม่ได้ทำการแคชซึ่ง 100 เท่า แต่ข้อดีของการทำดิสก์แคชซึ่งคือทำให้เวลาเฉลี่ยของการพบข้อมูลในหน่วยความจำดีขึ้น ซึ่งทำให้ประสิทธิภาพโดยรวมของระบบดีขึ้น (โดยทั่วไปจะเร็วขึ้น 100 เท่า สำหรับการ query ที่ถูกปรับอย่างดี)

3.2.2 ตั้งค่า DB_BLOCK_SIZE ให้สะท้อนถึงขนาดของการอ่านข้อมูล



รูปที่ 3-10 ความสัมพันธ์ระหว่าง DB_BLOCK_SIZE กับ DB_BLOCK_BUFFER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ดูแลระบบได้ปรับระบบแล้ว การดำเนินการใดๆ ไม่ควรกระทำ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในอรรถกถา แต่ละเทมเพลตสามารถมีขนาดของบล็อกที่ต่างกัน การมีขนาดของบล็อกที่ต่างกันไม่ใช่ข้อดี เพราะต้องมีการแบ่งหน่วยความจำแฉะ ขนาดต่าง ๆ ให้กับแต่ละบล็อกที่มีขนาดแตกต่างกันซึ่งขนาดของบล็อกต่าง ๆ ไม่ถูกทำให้ดีที่สุด(จะดีที่สุดสำหรับบล็อกที่ใช้ขนาดพื้นฐาน 'default') DB_BLOCK_SIZE ขนาดเล็กทำให้การนำข้อมูลอันเดียวกลับออกมาได้เร็ว และประหยัดส่วนที่เพิ่มเติมคือ ทำให้ Transactional concurrency ดีขึ้น (เพราะบล็อกที่ใช้ไม่มีข้อมูลที่ไมเกี่ยวข้องน้อย) และลดการขยายตัวของล็อกไฟล์ (Log file) เพราะช่องว่างในแต่ละบล็อกจะน้อยลง

3.2.3 DB_BLOCK_SIZE

ค่าดั้งเดิม : ขึ้นอยู่กับระบบปฏิบัติการ (มักเป็น 8192) : DEFAULT

ค่าที่เป็นไปได้ : ขึ้นอยู่กับระบบปฏิบัติการ

เป็นการกำหนดขนาดของแต่ละดาต้าเบส บล็อก บัฟเฟอร์(Database Block Buffer) ถึงแม้ว่าแถวของข้อมูลจะยาวหรือสั้น เราควรกำหนดขนาดเป็นค่าดั้งเดิม (Default) จะดีที่สุด นอกจากนี้ค่าพารามิเตอร์จะมีผลกระทบแค่เพียง ในกรณีที่เรามีการสร้างฐานข้อมูลในครั้งแรกเท่านั้น

3.2.4 DB_BLOCK_BUFFERS

ค่าดั้งเดิม : 32

ค่าที่เป็นไปได้ : 4 - ไม่จำกัดขอบเขต

เป็นการกำหนดขนาดบัฟเฟอร์ แฉะ (Buffer Cache) ของฐานข้อมูล ในหน่วยความจำเพื่อให้มีประสิทธิภาพในการทำงานดีขึ้น บัฟเฟอร์ แฉะจะทำการเก็บตาราง, ดัชนี (Index), คลัสเตอร์(Cluster) และ โรลแบ็ก เซกเมนต์(Rollback Segment)

เมื่อบัฟเฟอร์ แฉะมีขนาดสูงขึ้น จะทำให้กระบวนการที่เกี่ยวข้องกับหน่วยรับและแสดงผล(Input/Output) ลดลง และระบบจะการทำงานที่ดีขึ้น แต่ถ้าเมื่อใดที่เกิดการเพจจิง (Paging) และ สวอปปีง (Swapping) เกิดขึ้นมากเกินไปสำหรับโพรเซสผู้ใช้งานระบบหรือเอสจีเอ(SGA) เราจะต้องทำการลดขนาดของพารามิเตอร์ DB_BLOCK_BUFFERS เพื่อเพิ่มพื้นที่ในหน่วยความจำให้มากขึ้นกว่าเดิมคือเพิ่มพื้นที่ให้กับหน่วยความจำโดยรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 DB_FILES

ค่าดั้งเดิม : 200 (Default) ที่ติดตั้งมากับเครื่อง

ค่าที่เป็นไปได้ : 200 - ค่าของ MAXDATAFILES

เป็นการกำหนดจำนวนของไฟล์ฐานข้อมูลที่สามารถถูกเปิดขึ้นในเวลาพื้นฐานข้อมูลกำลังใช้งานอยู่ กำหนดค่านี้ต่ำกว่าค่าดั้งเดิมถ้าฐานข้อมูลมีไฟล์ข้อมูลอยู่ไม่ถึง 200 ไฟล์(เพื่อที่จะลดพื้นที่ที่ถูกใช้ในเอสจีเอ)

การเพิ่มค่านี้ทำได้โดยการหยุดใช้งานฐานข้อมูลก่อน(ShutDown) แล้วทำการเปลี่ยนค่าพารามิเตอร์ หลังจากนั้นทำการเริ่มต้นฐานข้อมูล (Starting Database) อีกครั้ง

3.2.6 LOG_BUFFER

ค่าดั้งเดิม : 4*ค่าของ DB_BLOCK_SIZE(8192)

ค่าที่เป็นไปได้ : ขึ้นอยู่กับระบบปฏิบัติการ

กำหนดจำนวน ไบต์(Byte) ของรีดูล็อก บัฟเฟอร์ในเอสจีเอ รีดูล็อกบัฟเฟอร์สามารถมีผลกระทบต่อประสิทธิภาพของระบบจัดการฐานข้อมูลเชิงวัตถุสัมพันธ์ (Relational Database Management System : RDBMS) โดยจะเก็บข้อมูลไว้ในบัฟเฟอร์ก่อนทำการเขียนข้อมูลลงสู่อีไดร์ฟ

ถ้ามีประสบการณ์เกี่ยวกับการติดขัดของหน่วยรับและแสดงผลข้อมูลบนดิสก์ ซึ่งจัดเก็บ รีดูล็อก ไฟล์ ให้ทำการเพิ่มจำนวนของพารามิเตอร์นี้ สำหรับโปรแกรมประยุกต์ที่ทำการเปลี่ยนแปลงข้อมูลบนฐานข้อมูลมาก ๆ ให้ทำการเพิ่มค่าพารามิเตอร์จาก 64K ไปเป็น 1 Mbyte จะทำให้ประสิทธิภาพดีขึ้น 7% - 10% แต่ถ้าเพิ่มขนาดเกิน 1 Mbyte ก็จะไม่ส่งผลให้เกิดการเปลี่ยนแปลงทางประสิทธิภาพมากนัก

ถ้าหากกำหนดพารามิเตอร์ให้มีค่ามากขึ้น จะทำให้ต้องใช้เนื้อที่ในหน่วยความจำของเอสจีเอมากขึ้นด้วย ถ้าระบบมีหน่วยความจำอย่างจำกัด อาจส่งผลให้เกิดการเพจจิง(Paging) และสวอปปีง(Swapping) ได้ เมื่อเป็นเช่นนี้ควรทำการลดค่าของพารามิเตอร์ลงเท่ากับค่าเดิม(Default) แต่ถ้ามีพื้นที่ในหน่วยความจำไม่จำกัด ให้ทำการกำหนดค่านี้เท่ากับ 1 Mbyte หลังจากนั้นทำการตรวจสอบประสิทธิภาพในเวลาต่อมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.7 LOG_CHECKPOINT_INTERVAL

ค่าตั้งเดิม : ขึ้นอยู่กับระบบปฏิบัติการ

ค่าที่เป็นไปได้ : 2 - ไม่จำกัดขอบเขต

กำหนดจำนวนบล็อกของ รีดู ล็อก ไฟล์ที่จะส่งผลให้เกิดการเช็คพอยท์ขึ้น โดยคำนึงอยู่เสมอว่า บล็อกเหล่านั้นเป็นบล็อกของระบบปฏิบัติการ ไม่ใช่บล็อกของออร์เกิล ถ้าหากกำหนดค่านี้น้อยกว่าขนาดของรีดู ล็อก ไฟล์ จะทำให้เกิดการเช็คพอยท์ที่บ่อยขึ้น ซึ่งเช็คพอยท์นี้เอง จะมีผลทำให้บล็อกบัฟเฟอร์ข้อมูลในส่วนที่มีแก๊ซ(Dirty Block) ถูกเขียนลงสู่ดิสก์ และจะเก็บตำแหน่งไว้บน รีดู ล็อก ไฟล์ ว่ามีการเกิดเช็คพอยท์ถึงไหนแล้วบนคอนโทรล ไฟล์(Control File) และไฟล์ฐานข้อมูล(Database File) แต่ให้คำนึงอยู่เสมอว่า ดาต้าเบสไรท์เตอร์จะนำดาต้าเบสบล็อกบัฟเฟอร์ในส่วนที่มีการแก๊ซ(Dirty Block) ลงสู่ดิสก์เมื่อใดก็ได้ โดยไม่ต้องคำนึงถึงการเช็คพอยท์ นั่นก็หมายความว่าตั้งแต่เกิดเช็คพอยท์ครั้งสุดท้ายนั้น ดาต้าเบสบล็อกบัฟเฟอร์ที่มีการแก๊ซอาจถูกเขียนลงสู่ฐานข้อมูลไปบ้างแล้วก็ได้

ถ้าต้องการ ให้การกู้ข้อมูลกลับคืนได้เร็วขึ้น(Fast Recovery Time) ควรกำหนดค่าของพารามิเตอร์ LOG_CHECKPOINT_INTERVAL ให้มีค่าที่ต่ำกว่าขนาดของ รีดู ล็อก ไฟล์ แต่ถ้าเรากำหนดถึงประสิทธิภาพการทำงานมาเป็นอันดับหนึ่ง ควรจะกำหนดค่าพารามิเตอร์นี้ให้เท่ากับหรือใหญ่กว่าขนาดของ รีดู ล็อก ไฟล์ เพื่อลดการเกิดเช็คพอยท์ และกิจกรรมของหน่วยรับและแสดงผลข้อมูลลงอีกด้วย

3.2.8 SHARED_POOL_SIZE

ค่าตั้งเดิม : 72 (MB)

ค่าที่เป็นไปได้ : ขึ้นอยู่กับค่า Maximum SGA Size* (MB)

กำหนดขนาดของแชร์ บัฟเฟอร์ พูล(Shared Buffer Pool) ในเอสซีเอรุ่น 10g (Version 10g) บัฟเฟอร์ในส่วนนี้จะเก็บบล็อกของเอสคิวแอล(SQL)และพีแอลเอสคิวแอล(PL/SQL) ที่สามารถใช้งานร่วมกันได้ และดาต้า ดิกชันนารี แคช รวมไปถึงในกรณีที่เป็นระบบผู้ให้บริการแบบมัลติทรีดเด็ค(Multi Threaded Server) ก็จะมีการเก็บข้อมูลเกี่ยวกับเซสชัน(Session) ด้วย

ควรที่จะมีการกำหนดขนาดของแชร์ พูล(Shared Pool) ให้ใหญ่เพียงพอ แต่ไม่ควรกำหนดให้ใหญ่เกินไป มิฉะนั้นจะทำให้สูญเสียพื้นที่ในหน่วยความจำไปโดยไม่จำเป็น แต่ถ้ามีการกำหนดค่าที่ต่ำเกินไป ก็จะทำให้ถึงข้อมูลบ่อยครั้งมากขึ้น เป็นการลดประสิทธิภาพการทำงานลงมากประมาณ 50 เปอร์เซ็นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก่อนที่จะทำการเพิ่มค่าพารามิเตอร์นี้ให้สูงขึ้น ควรมีการตรวจสอบก่อนว่า มีหน่วยความจำไม่เพียงพอหรือไม่ เพราะถ้ากำหนดค่านี้สูงกว่าจำนวนของหน่วยความจำที่ว่าง ก็จะทำให้เกิด การเพจจิง(Paging) และสวอปปี้ง(Swapping) ขึ้น ซึ่งเป็นการลดประสิทธิภาพการทำงานลง

ควรมีการปรับแต่งพารามิเตอร์ SHARED_POOL_SIZE ก่อนที่จะมีการปรับแต่งที่พารามิเตอร์ DB_BLOCK_BUFFERS เพราะว่าการกำหนดค่าของแชร์พูลที่ต่ำกว่าค่าที่ควรจะเป็น จะลดประสิทธิภาพการทำงานสูงมากกว่าการกำหนดค่าของ DB_BLOCK_BUFFERS ต่ำกว่าค่าที่เหมาะสม

สำหรับระบบที่มีขนาดเล็ก ๆ ควรจะทำการลดค่าดั้งเดิม(Default) ลงเหลือประมาณ MIN_SIZE = 68 เมกะไบต์(Mbytes) สำหรับระบบที่ใหญ่ขึ้นจนถึงปานกลาง ให้กำหนดเท่ากับค่าดั้งเดิมที่ดีที่สุด คือ CURRENT_SIZE = 72 เมกะไบต์(Mbytes) ส่วนระบบที่มีขนาดใหญ่ ให้ทำการเพิ่มขึ้นตามความเหมาะสมกับปริมาณงานที่ทำบ่อย ๆ เป็น เมกะไบต์(Mbytes) หลังจากนั้นควรที่จะมีการตรวจสอบประสิทธิภาพของแชร์ พูล แคช(Shared Pool Cache) บ่อย ๆ และปรับแต่งให้ค่าเหมาะสมในขณะนั้นด้วย

3.2.9 SORT_AREA_SIZE

ค่าดั้งเดิม : ขึ้นอยู่กับระบบปฏิบัติการ

ค่าที่เป็นไปได้ : ขึ้นอยู่กับระบบปฏิบัติการ

กำหนดขนาดในหน่วยไบต์(Byte) ซึ่ง โพรเซสของผู้ใช้งานระบบสามารถทำการจับจองเพื่อเรียงข้อมูลได้ ค่าพารามิเตอร์นี้สามารถกำหนดให้สูงกว่าค่าดั้งเดิมได้(65536 ไบต์) ถ้าหากว่าผู้ใช้บริการฐานข้อมูลมีพื้นที่หน่วยความจำมากพอ การปรับปรุงประสิทธิภาพการทำงานสามารถที่จะทำสลับกันไป ในระหว่างช่วงเวลาที่มิงานมากและงานน้อยได้ เช่น ในกรณีที่มีการทำรายงานในเวลา กลางคืน ก็จะมีการกำหนดพารามิเตอร์สูงขึ้นไปกว่าเดิม ที่เป็นเช่นนี้ก็เพราะว่า ในเวลา กลางคืนมีผู้ใช้ระบบน้อย ทำให้หน่วยความจำที่ว่างมีมากขึ้น จึงสามารถที่จะทำการเพิ่มค่าของ SORT_AREA_SIZE ได้ แต่จะต้องทำการกำหนดค่ากลับอย่างเดิมในเวลา กลางวันที่มีผู้ใช้ระบบมากขึ้นมีเช่นนั้นจะเป็นการลดประสิทธิภาพการทำงานลง

3.2.10 SORT_AREA_RETAINED_SIZE

ค่าดั้งเดิม : ค่าของ SORT_AREA_SIZE

ค่าที่เป็นไปได้ : 0 – ค่าของ SORT_AREA_SIZE

กำหนดขนาดในหน่วยไบต์ที่ออราเคิลจะทำการลดพื้นที่ในการเรียงข้อมูลลง ในกรณีที่เกี่ยวข้องกับการเรียงข้อมูลไม่ถูกอ้างถึง หน่วยความจำจะถูกทำให้ลดลงได้ในกรณีในทุก ๆ แถวถูกเฟตช์ (Fetch) แล้วจากพื้นที่ใช้ในการเรียงข้อมูล

3.2.11 SQL_TRACE

ค่าดั้งเดิม : FALSE

ค่าที่เป็นไปได้ : TRUE หรือ FALSE

กำหนดการใช้งานของเอสคิวแอล เทรซ (SQL Trace) ให้ทำงานได้หรือไม่ ค่าของ SQL_TRACE นี้ จะกำหนดให้เป็น FALSE เมื่อมีการสร้างฐานข้อมูลใหม่ มิเช่นนั้นจะทำให้การสร้างฐานข้อมูลช้าลง

ขอแนะนำว่าควรกำหนดค่าของ SQL_TRACE ให้เป็น FALSE จะดีที่สุด เพราะการกำหนดค่าให้เป็น TRUE จะใช้สำหรับการปรับแต่งฐานข้อมูลและใช้งานร่วมกับ TKPROF มิเช่นนั้นจะลดประสิทธิภาพการทำงานของออราเคิลลงมากกว่า 27 เปอร์เซ็นต์

3.2.12 LOG_CHECKPOINT_TIMEOUT

ค่าดั้งเดิม : 1800

ค่าที่เป็นไปได้ : ขึ้นอยู่กับระบบปฏิบัติการ

กำหนดให้เกิดเช็คพอยท์บ่อยมากขึ้น ถ้ากำหนดให้ค่าพารามิเตอร์นี้มากขึ้นกว่าค่าเดิม จะทำให้เกิดการเช็คพอยท์บ่อยมากขึ้นกว่าเดิม ซึ่งปกติเช็คพอยท์จะเกิดขึ้นเมื่อมีการเปลี่ยนล็อกสวิตช์ (Log Switch) ในเวลาที่รีด ล็อกไฟล์ ถูกเขียนจนเต็ม

LOG_CHECKPOINT_TIMEOUT นี้จะมีหน่วยเป็นวินาที นั่นก็คือเช็คพอยท์จะเกิดขึ้น อีกครั้งเป็นเวลาเท่ากับค่า LOG_CHECKPOINT_TIMEOUT วินาที จากเช็คพอยท์ครั้งสุดท้ายที่เกิดขึ้น

เพื่อให้ประสิทธิภาพในการทำงานของระบบฐานข้อมูลดีขึ้น ควรจะทำการกำหนดค่าพารามิเตอร์นี้เท่ากับ 1800 และค่าของ

LOG_CHECKPOINT_INTERVAL ให้มีขนาดใหญ่กว่าขนาดของ รีด ล็อก ไฟล์ ซึ่งจะทำให้เกิดเช็คพอยท์ขึ้น ในกรณีที่มีการเปลี่ยนรีด ล็อก ไฟล์เพื่อการเขียนข้อมูล

เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 ทำการทดลองวัดอัตราการพบและไม่พบข้อมูลบนไลบรารี แคช และ ดาต้า ดิกชันนารี แคช พารามิเตอร์ที่เกี่ยวข้องคือ SHARED_POOL_SIZE หน่วยเป็น [Byte]

4.1.1 ขั้นตอนการทดลอง

ขั้นแรกทำการเปลี่ยนค่าพารามิเตอร์ SHARED_POOL_SIZE ให้มีขนาดน้อย ๆ ก่อน โดยเริ่มที่ค่า 500,000 byte หลังจากนั้นก็ส่งคำสั่ง SQL Statement เข้าไปทำงานในระบบผ่านทาง SQL * Plus เพื่อดูอัตราการพบและไม่พบข้อมูลที่จะเกิดขึ้นเพราะว่าไลบรารี แคชจะเก็บคำสั่งเอสคิวเอล ที่ส่งเข้ามาทำงานในระบบ และเราจะดูผลได้จากโปรแกรมออรากเคิลเอนเตอร์ไพร์มาเนเจอร์(Oracle Enterprise Manager 9i) เป็นเครื่องมือที่ใช้จัดการฐานข้อมูล โดยทางออรากเคิลจะแนะนำให้เราลงโปรแกรมนี้หลังจากที่ติดตั้งฐานข้อมูลเสร็จแล้วไว้ใช้คู่กันเสมอ ผลการทดลองที่ได้จะแสดงในรูปแบบดังต่อไปนี้

Memory Efficiency

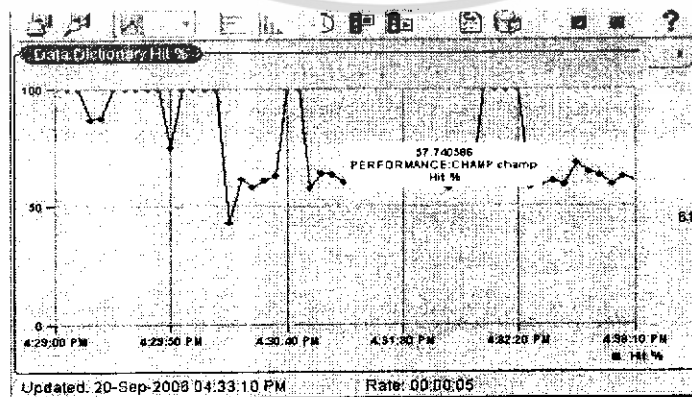
(Chart form was tabular, see data table below)

Instance Metrics	Buffer Cache Hit %	Library Cache Hit %	% Shared Pool Free
PERFORMANCE:CHAMP:champ	100.0	32.194046	31.389427

Buffer Cache Efficiency

(Chart form was tabular, see data table below)

รูปที่ 4-1 อัตราการพบไม่พบข้อมูลในไลบรารี แคชที่ค่า 500,000 byte



รูปที่ 4-2 อัตราการพบไม่พบข้อมูลในดาต้า ดิกชันนารี แคชที่ค่า 500,000 byte

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของบริษัทฯ เพื่อการศึกษารายงานนี้ เมื่อผู้ดูแลระบบได้แก้ไขข้อบกพร่องด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

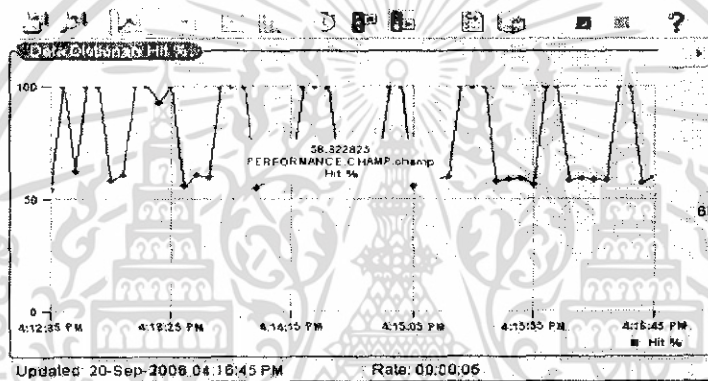
หลังจากนั้นทำการเพิ่มขนาดของแชร์พูลให้มีค่าเพิ่มขึ้นอีกเป็น 1,000,000 byte (1MB) ผลที่ได้ดังรูป

Memory Efficiency
(Chart form was tabular, see data table below)

Instance Metrics	Buffer Cache Hit %	Library Cache Hit %	% Shared Pool Free
PERFORMANCE:CHAMP:champ	100.0	32.420267	32.431471

Buffer Cache Efficiency

รูปที่ 4-3 อัตราการพบไม่พบข้อมูลในไลบรารี แคชที่ค่า 1,000,000 byte



รูปที่ 4-4 อัตราการพบไม่พบข้อมูลในตาต้า ดิกชันนารี แคชที่ค่า 1,000,000

และเพิ่มขนาดของแชร์พูลไปอีกสามค่าคือ [small], [medium], [Large] ที่ค่า 3,500,000 | 6,000,000 | 9,000,000 ตามลำดับ

4.1.2 ผลการทดลองวัดค่าอัตราการพบไม่พบข้อมูลบนไลบรารี แคช และตาต้า ดิกชันนารี แคช

ขนาด [BYTE] SHARED_POOL_SIZE	LIBRARY CACHE		DATA DICTIONARY CACHE	
	HIT RATIO	MISS RATIO	HIT RATIO	MISS RATIO
500,000	82.29 %	17.81 %	57.74 %	42.26 %
1,000,000	82.42 %	17.58 %	58.32 %	41.68 %
3,500,000 [SMALL]	89.74 %	10.26 %	91.30 %	8.70 %
6,000,000 [MEDIUM]	94.12 %	5.88 %	92.50 %	7.50 %
9,000,000 [LARGE]	96.31 %	3.69 %	95.00 %	5.00 %

ตารางที่ 4-1 ผลการทดลองวัดค่าอัตราการพบไม่พบข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองเปรียบเทียบการเข้าถึงข้อมูลแบบใช้อินเด็ก กับ ไม่ใช้อินเด็ก

4.2.1 ขั้นตอนการทดลอง

ขั้นแรกทำการสร้างเทเบิลขึ้นมาในการทดลองนี้ เทเบิลชื่อ LICEN คือเทเบิลที่เก็บข้อมูลใบขับขี่รถยนต์และรถจักรยานยนต์ โดยได้ทำการสร้างอินเด็กบนคอลัมภ์ LNUM ชื่อว่า TEST1_PK โดยเพิ่มข้อมูลเข้าไป 120 เร็คคอร์ด หลังจากนั้นทำการ คิวรี่ (Query) ข้อมูลทั้งหมดออกมาโดยใช้คำสั่งดังรูปที่ 4-5

```
SQL> select * from licen;
```

LNUM	TYPE	NAME	ID	PROVINCE
112	pickup	chinjang	37015315	janthaburi
113	truck	angle	37015316	sahaigolok
114	car	weezer	37015317	yala
115	car	evan	37015318	narativas
116	car	adam	37015319	saraburi
117	truck	tong	37015320	RATCHBURI
118	truck	ton	37015321	RATCHBURI
119	motorcycles	pratana	37015322	chonburi
220	motrocycles	hellboy	37015323	patchchaburi
1	MOTORCYCLE	SATIT	47001320	BANGKOK
2	MOTORCYCLE	SUDARAT	47001321	SURATCH

รูปที่ 4-5 ข้อมูลทั้งหมดของเทเบิล LICEN

หลังจากนั้นใช้คำสั่ง ANALYZE TABLE [table name] COMPUTE STATISTICS FOR TABLE FOR ALL INDEXES; เพื่อเก็บสถิติและตามด้วยคำสั่ง SET AUTOTRACE [ON|OFF|ONLY] [EXPLAIN|STATISTICS]; ซึ่งเป็นความสามารถของออราเคิล เพื่อที่หลังจากการคิวรี่(QUERY) มันจะนำแผนงานและสถิติในแต่ละครั้งออกมาแสดงโดยอัตโนมัติ ดังรูป ต่อไปนี้

```
SQL> set autotrace on explain; (คำสั่งที่ใช้ดูแผนการรัน)
```

```
SQL> select name, province  
2 from licen  
3 where province = 'ubon';
```

แผนการรัน

Plan hash value: 1888635767

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	21	3 (0)	00:00:01
* 1	TABLE ACCESS FULL	LICEN	1	21	3 (0)	00:00:01

รูปที่ 4-6 แผนการรันเมื่อไม่เลือกใช้อินเด็กซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนนี้ทำการสร้างอินเด็กเพิ่มเข้าไปบนคอลัมภ์ I.NUM จะได้แผนงานดังรูปที่ 4-7

แผนการรัน

Plan hash value: 4088760038

```

-----
| Id | Operation | Name | Rows | Bytes | Cost (%CPU)| Time
-----
| 0 | SELECT STATEMENT | | | | | |
:00:01 |
| 1 | TABLE ACCESS BY INDEX ROWID | LICEN | 1 | 35 | 2 (0)| 00
:00:01 |
|* 2 | INDEX RANGE SCAN | TEST1_PK | 1 | | 1 (0)| 00
:00:01 |
-----

```

รูปที่ 4-7 แผนการรันเมื่อเลือกใช้อินเด็กซ์

4.2.2 ผลการทดลองเปรียบเทียบอัตราการใช้ COST และ TIME จากแผนงานทั้งสองซึ่งได้ค่าที่แตกต่างกันหมายความว่าประสิทธิภาพแตกต่างกัน

OPERATION	COST (%CPU)		TIME	
	INDEX	NO_INDEX	INDEX	NO_INDEX
SELECT STATEMENT	2	3	00:01	00:01
TABLE ACCESS FULL(Full table scan)	-	3	00:01	00:01
TABLE ACCESS BY INDEX ROWID	2	-	00:01	00:01
INDEX RANGE SCAN(Test1_Pk)	1	-	00:01	00:01

ตารางที่ 4-2 ผลการทดลองเปรียบเทียบอัตราการใช้ COST และ TIME

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลอง B-tree index และ Bitmap index

4.3.1 ขั้นตอนการทดลอง

4.3.1.1 ใช้คำสั่ง SET AUTOTRACE ON EXPLAIN; ในโปรแกรม SQL*Plus เพื่อให้ SQL*Plus แสดงแผนกาทำงานของคำสั่ง SQL

4.3.1.2 ทำการค้นหาข้อมูลในตาราง borrow ขณะที่ยังไม่มีอินเด็กซ์ที่คอลัมน์ student_id โดยใช้คำสั่ง

```
SELECT      COUNT(*)
FROM        BORROW
WHERE       STUDENT_ID = '500000';
```

แผนการรัน

Plan hash value: 1206838491

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	7	911 (5)	00:00:11
1	SORT AGGREGATE		1	7		
* 2	TABLE ACCESS FULL	BORROW	2	14	911 (5)	00:00:11

Predicate Information (identified by operation id):

2 - filter("STUDENT_ID"='500000')

รูปที่ 4-8 แผนการรันที่ 1

4.3.1.3 ทำการสร้าง B-tree อินเด็กซ์บนคอลัมน์ student_id ด้วยคำสั่ง CREATE INDEX ID1 ON BORROW(STUDENT_ID); จากนั้นทำการค้นหาข้อมูลในตาราง borrow ขณะที่มีอินเด็กซ์ ID1 ตัวเดียวเท่านั้นที่อินเด็กซ์บนคอลัมน์ student_id โดยใช้คำสั่ง

```
SELECT      COUNT(*)
FROM        BORROW
WHERE       STUDENT_ID = '500000';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนการสําน

Plan hash value: 1073521204

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	7	3 (0)	00:00:01
1	SORT AGGREGATE		1	7		
* 2	INDEX RANGE SCAN	ID1	2	14	3 (0)	00:00:01

Predicate Information (identified by operation id):

2 - access("STUDENT_ID"='500000')

รูปที่ 4-9 แผนการรันที่ 2

4.3.1.4 ทำการสร้าง Bitmap อินเด็กซ์บนคอลัมน์ student_id ด้วยคำสั่ง CREATE BITMAP INDEX ID2 ON BORROW(STUDENT_ID); จากนั้นทำการค้นหาข้อมูลในตาราง borrow ขณะที่มีอินเด็กซ์ ID2 ตัวเดียวเท่านั้นที่อินเด็กซ์บนคอลัมน์ student_id โดยใช้คำสั่ง

```
SELECT COUNT(*)
FROM BORROW
WHERE STUDENT_ID = '500000';
```

แผนการสําน

Plan hash value: 1242517771

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	7	3 (0)	00:00:01
1	SORT AGGREGATE		1	7		
2	BITMAP CONVERSION COUNT		2	14	3 (0)	00:00:01
* 3	BITMAP INDEX SINGLE VALUE	ID2				

Predicate Information (identified by operation id):

3 - access("STUDENT_ID"='500000')

รูปที่ 4-10 แผนการรันที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1.5 ทำการค้นหาข้อมูลในตาราง student ขณะที่ยังไม่มีอินเด็กซ์ที่คอลัมน์

en_name โดยใช้คำสั่ง

```
SELECT      COUNT(*)
FROM        STUDENT
WHERE       EN_NAME = 'Michael';
```

แผนการรัน

Plan hash value: 4030891848

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	8	9352 (2)	00:01:53
1	SORT AGGREGATE		1	8		
* 2	TABLE ACCESS FULL	STUDENT	2315	18520	9352 (2)	00:01:53

Predicate Information (identified by operation id):

2 - filter("EN_NAME"='Michael')

รูปที่ 4-11 แผนการรันที่ 4

4.3.1.6 ทำการสร้าง B-tree อินเด็กซ์บนคอลัมน์ en_name ด้วยคำสั่ง CREATE

INDEX ID3 ON STUDENT(EN_NAME); จากนั้นทำการค้นหาข้อมูลในตาราง student ขณะที่ไม่มีอินเด็กซ์ ID3 ตัวเดียวเท่านั้นที่อินเด็กซ์บนคอลัมน์ en_name โดยใช้คำสั่ง

```
SELECT      COUNT(*)
FROM        BORROW
WHERE       STUDENT_ID = '500000';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนการรับ

Plan hash value: 2209262735

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	8	9 (0)	00:00:01
1	SORT AGGREGATE		1	8		
* 2	INDEX RANGE SCAN	ID3	2315	18520	9 (0)	00:00:01

Predicate Information (identified by operation id):

2 - access('EN_NAME'='Michael')

รูปที่ 4-12 แผนการรับที่ 5

4.3.1.7 ทำการสร้าง Bitmap อินเด็กซ์บนคอลัมน์ en_name ด้วยคำสั่ง CREATE BITMAP INDEX ID4 ON STUDENT(EN_NAME); จากนั้นทำการค้นหาข้อมูลในตาราง student ขณะที่อินเด็กซ์ ID4 ตัวเดียวเท่านั้นที่อินเด็กซ์บนคอลัมน์ en_name โดยใช้คำสั่ง

```
SELECT COUNT(*)
FROM BORROW
WHERE STUDENT_ID = '500000';
```

แผนการรับ

Plan hash value: 1578375609

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	8	1 (0)	00:00:01
1	SORT AGGREGATE		1	8		
2	BITMAP CONVERSION COUNT		2315	18520	1 (0)	00:00:01
* 3	BITMAP INDEX SINGLE VALUE	ID4				

Predicate Information (identified by operation id):

3 - access('EN_NAME'='Michael')

รูปที่ 4-13 แผนการรับที่ 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.1.8 ทำการค้นหาข้อมูลในตาราง student ขณะที่ยังไม่มีอินเด็กซ์ที่คอลัมน์

faculty id โดยใช้คำสั่ง

```
SELECT      COUNT(*)
FROM        STUDENT
WHERE       FACULTY_ID = '1';
```

แผนการรัน

Plan hash value: 4030891848

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	3	9327 (1)	00:01:52
1	SORT AGGREGATE		1	3		
* 2	TABLE ACCESS FULL	STUDENT	100K	292K	9327 (1)	00:01:52

Predicate Information (identified by operation id):

2 - filter("FACULTY_ID"='1')

รูปที่ 4-14 แผนการรันที่ 7

4.3.1.9 ทำการสร้าง B-tree อินเด็กซ์บนคอลัมน์ faculty_id ด้วยคำสั่ง CREATE

INDEX ID5 ON STUDENT(FACULTY_ID); จากนั้นทำการค้นหาข้อมูลในตาราง student ขณะที่มีอินเด็กซ์ ID5 ตัวเดียวเท่านั้นที่อินเด็กซ์บนคอลัมน์ faculty_id โดยใช้คำสั่ง

```
SELECT      COUNT(*)
FROM        STUDENT
WHERE       FACULTY_ID = '1';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนการรับ

Plan hash value: 2442285474

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	3	189 (3)	00:00:03
1	SORT AGGREGATE		1	3		
* 2	INDEX RANGE SCAN	ID5	100K	292K	189 (3)	00:00:03

Predicate Information (identified by operation id):

2 - access("FACULTY_ID"='1')

รูปที่ 4-15 แผนการรับที่ 8

4.3.1.10 ทำการสร้าง Bitmap อินเด็กซ์บนคอลัมน์ faculty_id ด้วย CREATE BITMAP INDEX ID6 ON STUDENT(FACULTY_ID); จากนั้นทำการค้นหาข้อมูลในตาราง student ขณะที่มีอินเด็กซ์ ID6 ตัวเดียวเท่านั้นที่อินเด็กซ์บนคอลัมน์ faculty_id โดยใช้คำสั่ง

```
SELECT COUNT(*)
FROM STUDENT
WHERE FACULTY_ID='1';
```

แผนการรับ

Plan hash value: 524931997

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	3	20 (0)	00:00:01
1	SORT AGGREGATE		1	3		
2	BITMAP CONVERSION COUNT		100K	292K	20 (0)	00:00:01
* 3	BITMAP INDEX SINGLE VALUE	ID6				

Predicate Information (identified by operation id):

3 - access("FACULTY_ID"='1')

รูปที่ 4-16 แผนการรับที่ 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 ผลการทดลองเปรียบเทียบค่า Cost จากแผนการรันที่ 1-9

Table name	borrow	student	student
Distinct Keys	632740	431	10
Column	student_id	en_name	faculty_id
ไม่มี index	แผนการรันที่ 1 Cost = 911	แผนการรันที่ 4 Cost = 9352	แผนการรันที่ 7 Cost = 9327
B-tree	แผนการรันที่ 2 Cost = 3	แผนการรันที่ 5 Cost = 9	แผนการรันที่ 8 Cost = 189
Bit map	แผนการรันที่ 3 Cost = 3	แผนการรันที่ 6 Cost = 1	แผนการรันที่ 9 Cost = 20

ตารางที่ 4-3

4.4 การทดลองเรื่องการ Join

4.4.1 ขั้นตอนการทดลองของ

4.4.1.1 ใช้คำสั่ง SET AUTOTRACE ON EXPLAIN; ในโปรแกรม SQL*Plus เพื่อทำให้ SQL*Plus แสดงแผนกาทำงานของคำสั่ง SQL

4.4.1.2 ทำการ Join ตาราง book กับตาราง borrow โดยใช้คำสั่ง

```
SELECT B.BOOK_ID, B.BK_NAME, BOR.BSTATUS
FROM BOOK B, BORROW BOR
WHERE B.BOOK_ID = BOR.BOOK_ID AND B.BOOK_ID < 100;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนการรับ

Plan hash value: 3785053752

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		54293	1802K	1351 (6)	00:00:17
* 1	HASH JOIN		54293	1802K	1351 (6)	00:00:17
* 2	TABLE ACCESS FULL	BOOK	5000	126K	420 (2)	00:00:06
3	TABLE ACCESS FULL	BORROW	1000K	7812K	912 (5)	00:00:11

predicate Information (identified by operation id):

- 1 - access("B"."BOOK_ID"="BOR"."BOOK_ID")
- 2 - filter(TO_NUMBER("B"."BOOK_ID")<100)

รูปที่ 4-17 Hash Join

4.4.1.2 ทำการ Join ตาราง book กับตาราง borrow โดยใช้คำสั่ง

```
SELECT /*+ORDERED*/ B.BOOK_ID, B.BK_NAME, BOR.BSTATUS
FROM BORROW BOR, BOOK B
WHERE B.BOOK_ID = BOR.BOOK_ID AND B.BOOK_ID < 100;
```

แผนการรับ

Plan hash value: 841347714

Id	Operation	Name	Rows	Bytes	TempSpc	Cost (%CPU)	Time
0	SELECT STATEMENT		54293	1802K		2317 (4)	00:00:28
* 1	HASH JOIN		54293	1802K	19M	2317 (4)	00:00:28
2	TABLE ACCESS FULL	BORROW	1000K	7812K		912 (5)	00:00:11
* 3	TABLE ACCESS FULL	BOOK	5000	126K		420 (2)	00:00:06

predicate Information (identified by operation id):

- 1 - access("B"."BOOK_ID"="BOR"."BOOK_ID")
- 3 - filter(TO_NUMBER("B"."BOOK_ID")<100)

รูปที่ 4-18 Hash Join

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง

5.1 สรุปผลการทดลองเปรียบเทียบการเข้าถึงข้อมูลแบบใช้อินเด็ก กับ ไม่ใช้อินเด็ก(อย่างง่าย)

จากการทดลองในรูปแบบที่ 4-6 จะเห็นว่า Optimizer ของ ออราเคิล เลือกแผนงาน ที่แตกต่างกับรูปที่ 4-7 ซึ่งประสิทธิภาพ อาจแตกต่างกันขึ้นอยู่กับขนาดของแต่ละเทเบิล อย่างไรก็ตาม ตัวอย่างข้างบนนั้นเป็นการใช้ RBO (Rules Based Optimizer) ซึ่งไม่ฉลาดนัก หากเราใช้ CBO (Cost Based Optimizer) ดังรูปที่ 4-7 โปรแกรม จะฉลาดมากพอที่จะรู้ว่าแผนที่ดีที่สุด (ซึ่งจะเป็นแผนเดียวกัน เพราะ output ที่ได้ไม่แตกต่างกัน) คืออะไร และเลือกแผนนั้นให้เรา

COST เป็นตัวเลขที่ ออราเคิล คำนวณออกมาเพื่อใช้ในการเปรียบเทียบว่าแผนงานใดจะมีประสิทธิภาพในการทำงานสูงสุด (COST น้อยที่สุด)

5.2 สรุปผลการทดลอง B-tree index และ Bitmap index

เมื่อ 2 ตารางมี จำนวน row เท่ากัน ตารางที่มีขนาดใหญ่กว่าจะมีค่า cost จากการทำ full table scan มากกว่า

B-tree index เหมาะกับการทำอินเด็กซ์บนคอลัมน์ที่มีจำนวน Distinct key มากๆ มากกว่า Bitmap index เพราะในกรณีนี้ B-tree index ทำให้คำสั่ง INSERT เร็วกว่า Bitmap index

Bitmap index เหมาะกับการทำอินเด็กซ์บนคอลัมน์ที่มีจำนวน Distinct key ไม่มาก ค่า cost ของคำสั่ง SELECT ในกรณีนี้ Bitmap index จะน้อยกว่า B-tree index เพราะขนาดของ Bitmap index เล็กกว่า B-tree index

5.3 สรุปผลการทดลอง การ Join

การทำ Hash join Optimizer จะเลือกผลลัพธ์ที่ได้จากตารางที่มีค่าน้อยกว่า มาทำการสร้าง hash table ในหน่วยความจำ ค่า cost จากการนำผลลัพธ์ที่มีค่าน้อย(ผลลัพธ์จากตาราง book)มาสร้าง hash table จะมีค่าน้อยกว่าการนำผลลัพธ์ที่มีค่ามาก(ผลลัพธ์จากตาราง borrow)มาสร้าง hash table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. การจัดการระบบฐานข้อมูลอย่างมืออาชีพ (Oracle DBA) ผู้แต่ง ญาณี กาชัย

ISBN: 974-91140-6-X

2. สร้างระบบฐานข้อมูลด้วย PL/SQL และ Oracle Developer ผู้แต่ง ยุทธรา ติลาศวัฒน์กุล
อมรพันธุ์ คำอรรถ yudhana@yahoo.com ISBN: 947-619-03-95-4

3. Oracle 9i Performance Tuning Tips & Techniques Richard J. Niemiec

ISBN: 0-07-222473-8

McGraw-Hill/Osborne

2100 Powell Street, 10th Floor

Emeryville, California 94608

U. S. A.

4. Database Performance Tuning and Optimization Using Oracle

ISBN: 0-387-953930-0

Sitansu S. Mitra

Senior Principal Engineer and Database Management Specialist

Computer Sciences Corporation

5 Cambridge, MA 02139

USA

mitra4aol.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

พารามิเตอร์ทั้งหมดของออร่าเคิลรุ่น 10g

NAME	VALUE	ISMODIFIED
log_archive_trace	0	FALSE
log_buffer	2899456	FALSE
log_checkpoint_interval	0	FALSE
log_checkpoint_timeout	1800	FALSE
log_checkpoints_to_alert	FALSE	FALSE
log_file_name_convert	-	FALSE
logmnr_max_persistent_sessions	1	FALSE
max_commit_propagation_delay	0	FALSE
max_dispatchers	-	FALSE
max_dump_file_size	UNLIMITED	FALSE
max_enabled_roles	150	FALSE
max_shared_servers	-	FALSE
nls_calendar	-	FALSE
nls_comp	-	FALSE
nls_currency	-	FALSE
nls_date_format	-	FALSE
nls_date_language	-	FALSE
nls_dual_currency	-	FALSE
nls_iso_currency	-	FALSE
nls_language	AMERICAN	FALSE
nls_length_semantics	BYTE	FALSE
nls_nchar_conv_excp	FALSE	FALSE
nls_numeric_characters	-	FALSE
nls_sort	-	FALSE
nls_territory	AMERICAN	FALSE
nls_time_format	-	FALSE
nls_time_tz_format	-	FALSE
object_cache_max_size_percent	10	FALSE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NAME	VALUE	ISMODIFIED
object_cache_optimal_size	102400	FALSE
olap_page_pool_size	0	FALSE
open_cursors	300	FALSE
open_links	4	FALSE
open_links_per_instance	4	FALSE
optimizer_dynamic_sampling	2	FALSE
optimizer_features_enable	10.2.0.1	FALSE
optimizer_index_caching	0	FALSE
optimizer_index_cost_adj	100	FALSE
optimizer_mode	ALL_ROWS	FALSE
optimizer_secure_view_merging	TRUE	FALSE
os_authent_prefix	OP\$S	FALSE
os_roles	FALSE	FALSE
parallel_adaptive_multi_user	TRUE	FALSE
parallel_automatic_tuning	FALSE	FALSE
parallel_execution_message_size	2148	FALSE
parallel_instance_group	-	FALSE
parallel_max_servers	20	FALSE
parallel_min_percent	0	FALSE
parallel_server	FALSE	FALSE
parallel_server_instances	1	FALSE
parallel_threads_per_cpu	2	FALSE
pga_aggregate_target	16777216	FALSE
plsql_ccflags	-	FALSE
plsql_code_type	INTERPRETED	FALSE
plsql_compiler_flags	INTERPRETED, NON_DEBUG	FALSE
plsql_debug	FALSE	FALSE
plsql_native_library_dir	-	FALSE
plsql_native_library_subdir_count	0	FALSE
plsql_optimize_level	2	FALSE
plsql_v2_compatibility	FALSE	FALSE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NAME	VALUE	ISMODIFIED
plsql_warnings	DISABLE:ALL	FALSE
pre_page_sga	FALSE	FALSE
processes	150	FALSE
query_rewrite_enabled	TRUE	FALSE
query_rewrite_integrity	enforced	FALSE
rdbms_server_dn	-	FALSE
read_only_open_delayed	FALSE	FALSE
recovery_parallelism	0	FALSE
recyclebin	on	FALSE
remote_archive_enable	true	FALSE
remote_dependencies_mode	TIMESTAMP	FALSE
remote_listener	-	FALSE
remote_login_passwordfile	EXCLUSIVE	FALSE
remote_os_authent	FALSE	FALSE
remote_os_roles	FALSE	FALSE
replication_dependency_tracking	TRUE	FALSE
resource_limit	FALSE	FALSE
resource_manager_plan	-	FALSE
resumable_timeout	0	FALSE
rollback_segments	-	FALSE
serial_reuse	disable	FALSE
service_names	orcl	FALSE
session_cached_cursors	20	FALSE
session_max_open_files	10	FALSE
sessions	170	FALSE
sga_max_size	167772160	FALSE
sga_target	167772160	FALSE
shadow_core_dump	partial	FALSE
shared_memory_address	0	FALSE
shared_pool_reserved_size	3984588	FALSE
shared_pool_size	0	FALSE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NAME	VALUE	ISMODIFIED
shared_server_sessions	-	FALSE
shared_servers	1	FALSE
skip_unusable_indexes	TRUE	FALSE
smtp_out_server	-	FALSE
sort_area_retained_size	0	FALSE
sort_area_size	65536	FALSE
spfile	F:\ORACLE\PRODUCT\10.2.0\NDB_1\NDBS\SPFILEORCL.ORA	FALSE
sql92_security	FALSE	FALSE
sql_trace	FALSE	FALSE
sql_version	NATIVE	FALSE
sqltune_category	DEFAULT	FALSE
standby_archive_dest	%ORACLE_HOME%\RDBMS	FALSE
standby_file_management	MANUAL	FALSE
star_transformation_enabled	FALSE	FALSE
statistics_level	TYPICAL	FALSE
streams_pool_size	0	FALSE
tape_asynch_io	TRUE	FALSE
thread	0	FALSE
timed_os_statistics	0	FALSE
timed_statistics	TRUE	FALSE
trace_enabled	TRUE	FALSE
tracefile_identifier	-	FALSE
transactions	187	FALSE
transactions_per_rollback_segment	5	FALSE
undo_management	AUTO	FALSE
undo_retention	900	FALSE
undo_tablespace	UNDOTBS1	FALSE
use_indirect_data_buffers	FALSE	FALSE
user_dump_dest	E:\ORACLE\PRODUCT\10.2.0\ADMIN\ORCL\UDUMP	FALSE
utl_file_dir	-	FALSE
workarea_size_policy	AUTO	FALSE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

วิว(V\$Views) ทั้งหมดของออราเคิลรุ่น 10g

V\$ACCESS	V\$ACTIVE_INSTANCES
V\$ACTIVE_SERVICES	V\$ACTIVE_SESSION_HISTORY
V\$ACTIVE_SESS_POOL_MTH	V\$ADVISOR_PROGRESS
VSALERT_TYPES	V\$AQI
V\$ARCHIVE	V\$ARCHIVED_LOG
V\$ARCHIVE_DEST	V\$ARCHIVE_DEST_STATUS
V\$ARCHIVE_GAP	V\$ARCHIVE_PROCESSES
V\$ASM_ALIAS	V\$ASM_CLIENT
V\$ASM_DISK	V\$ASM_DISKGROUP
V\$ASM_DISKGROUP_STAT	V\$ASM_DISK_STAT
V\$ASM_FILE	V\$ASM_OPERATION
V\$ASM_TEMPLATE	V\$AW_AGGREGATE_OP
V\$AW_ALLOCATE_OP	V\$AW_CALC
V\$AW_LONGOPS	V\$AW_OLAP
V\$AW_SESSION_INFO	V\$BACKUP
V\$BACKUP_ARCHIVELOG_DETAILS	V\$BACKUP_ARCHIVELOG_SUMMARY
V\$BACKUP_ASYNC_IO	V\$BACKUP_CONTROLFILE_DETAILS
V\$BACKUP_CONTROLFILE_SUMMARY	V\$BACKUP_COPY_DETAILS
V\$BACKUP_COPY_SUMMARY	V\$BACKUP_CORRUPTION
V\$BACKUP_DATAFILE	V\$BACKUP_DATAFILE_DETAILS
V\$BACKUP_DATAFILE_SUMMARY	V\$BACKUP_DEVICE
V\$BACKUP_PIECE	V\$BACKUP_PIECE_DETAILS
V\$DATAFILE	V\$DATAFILE_COPY
V\$DATAFILE_HEADER	V\$DATAGUARD_CONFIG
V\$DATAGUARD_STATS	V\$DATAGUARD_STATUS
V\$DATAPUMP_JOB	V\$DATAPUMP_SESSION
V\$DBFILE	V\$DBLINK
V\$DB_CACHE_ADVICE	V\$DB_OBJECT_CACHE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

V\$DB_PIPES	V\$DB_TRANSPORTABLE_PLATFORM
V\$DELETED_OBJECT	V\$DISPATCHER
V\$DISPATCHER_CONFIG	V\$DISPATCHER_RATE
V\$DLM_ALL_LOCKS	V\$DLM_CONVERT_LOCAL
V\$DLM_CONVERT_REMOTE	V\$DLM_LATCH
V\$DLM_LOCKS	V\$DLM_MISC
V\$DLM_RESS	V\$DLM_TRAFFIC_CONTROLLER
V\$ENABLEDPRIVS	V\$ENQUEUE_LOCK
V\$ENQUEUE_STAT	V\$ENQUEUE_STATISTICS
V\$EVENTMETRIC	V\$EVENT_HISTOGRAM
V\$EVENT_NAME	V\$EXECUTION
V\$FAST_START_SERVERS	V\$FAST_START_TRANSACTIONS
V\$FILEMETRIC	V\$FILEMETRIC_HISTORY
V\$FILESPACE_USAGE	V\$FILESTAT
V\$FILE_CACHE_TRANSFER	V\$FILE_HISTOGRAM
V\$FILE_PING	V\$FIXED_TABLE
V\$FIXED_VIEW_DEFINITION	V\$FLASHBACK_DATABASE_LOG
V\$FLASHBACK_DATABASE_LOGFILE	V\$FLASHBACK_DATABASE_STAT
V\$FLASH_RECOVERY_AREA_USAGE	V\$GCSHVMaster_INFO
V\$GCSPFMASTER_INFO	V\$GC_ELEMENT
V\$GC_ELEMENTS_WITH_COLLISIONS	V\$GES_BLOCKING_ENQUEUE
V\$GES_ENQUEUE	V\$GLOBALCONTEXT
V\$GLOBAL_BLOCKED_LOCKS	V\$GLOBAL_TRANSACTION
V\$HS_AGENT	V\$HS_PARAMETER
V\$HS_SESSION	V\$HVMaster_INFO
V\$INDEXED_FIXED_COLUMN	V\$INSTANCE
V\$INSTANCE_CACHE_TRANSFER	V\$INSTANCE_LOG_GROUP
V\$INSTANCE_RECOVERY	V\$JAVAPOOL
V\$JAVA_LIBRARY_CACHE_MEMORY	V\$JAVA_POOL_ADVICE
V\$LATCH	V\$LATCHHOLDER
V\$LATCHNAME	V\$LATCH_CHILDREN
V\$LATCH_MISSES	V\$LATCH_PARENT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

V\$LIBRARYCACHE	V\$LIBRARY_CACHE_MEMORY
V\$LICENSE	V\$LOADISTAT
VSLOADPSTAT	VSLOCK
V\$LOCKED_OBJECT	V\$LOCKS_WITH_COLLISIONS
V\$LOCK_ACTIVITY	VSLOCK_ELEMENT
V\$LOCK_TYPE	V\$LOG
V\$LOGFILE	V\$LOGHIST
VSLOGMNR_CALLBACK	VSLOGMNR_CONTENTS
VSLOGMNR_DICTIONARY	V\$LOGMNR_DICTIONARY_LOAD
V\$LOGMNR_LATCH	V\$LOGMNR_LOGFILE
VSLOGMNR_LOGS	V\$LOGMNR_PARAMETERS
VSLOGMNR_PROCESS	V\$LOGMNR_REGION
VSLOGMNR_SESSION	V\$LOGMNR_STATS
VSLOGMNR_TRANSACTION	VSLOGSTDBY
V\$LOGSTDBY_PROCESS	V\$LOGSTDBY_PROGRESS
V\$LOGSTDBY_STATE	V\$LOGSTDBY_STATS
V\$LOGSTDBY_TRANSACTION	V\$LOG_HISTORY
V\$MANAGED_STANDBY	V\$MAP_COMP_LIST
V\$MAP_ELEMENT	V\$MAP_EXT_ELEMENT
V\$MAP_FILE	V\$MAP_FILE_EXTENT
V\$MAP_FILE_IO_STACK	V\$MAP_LIBRARY
V\$MAP_SUBELEMENT	V\$MAX_ACTIVE_SESS_TARGET_MTH
V\$METRIC	V\$METRICGROUP
V\$METRICNAME	V\$METRIC_HISTORY
V\$MTTR_TARGET_ADVICE	V\$MUTEX_SLEEP
V\$MUTEX_SLEEP_HISTORY	V\$MVREFRESH
V\$MYSTAT	V\$NLS_PARAMETERS
V\$NLS_VALID_VALUES	V\$OBJECT_DEPENDENCY
V\$OBSOLETE_PARAMETER	V\$OFFLINE_RANGE
V\$OPEN_CURSOR	V\$OPTION
V\$OSSTAT	V\$PARALLEL_DEGREE_LIMIT_MTH
V\$PARAMETER	V\$PARAMETER2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

V\$PARAMETER_VALID_VALUES	V\$PGASTAT
V\$PGA_TARGET_ADVICE	V\$PGA_TARGET_ADVICE_HISTOGRAM
V\$PQ_SESSTAT	V\$PQ_SLAVE
V\$PQ_SYSSTAT	V\$PQ_TQSTAT
V\$PROCESS	V\$PROCESS_MEMORY
V\$PROCESS_MEMORY_DETAIL	V\$PROCESS_MEMORY_DETAIL_PROG
V\$PROPAGATION_RECEIVER	V\$PROPAGATION_SENDER
V\$PROXY_ARCHIVEDLOG	V\$PROXY_ARCHIVELOG_DETAILS
V\$PROXY_ARCHIVELOG_SUMMARY	V\$PROXY_COPY_DETAILS
V\$PROXY_COPY_SUMMARY	V\$PROXY_DATAFILE
V\$PWFILERS	V\$PX_BUFFER_ADVICE
V\$PX_PROCESS	V\$PX_PROCESS_SYSSTAT
V\$PX_SESSION	V\$PX_SESSTAT
V\$QUEUE	V\$QUEUEING_MTH
V\$RECOVERY_FILE_DEST	V\$RECOVERY_FILE_STATUS
V\$RECOVERY_LOG	V\$RECOVERY_PROGRESS
V\$RECOVERY_STATUS	V\$RECOVER_FILE
V\$REPLPROP	V\$REPLQUEUE
V\$REQDIST	V\$RESERVED_WORDS
V\$RESOURCE	V\$RESOURCE_LIMIT
V\$RESTORE_POINT	V\$RESUMABLE
V\$RFS_THREAD	V\$RMAN_BACKUP_JOB_DETAILS
V\$RMAN_BACKUP_SUBJOB_DETAILS	V\$RMAN_BACKUP_TYPE
V\$RMAN_CONFIGURATION	V\$RMAN_ENCRYPTION_ALGORITHMS
V\$RMAN_OUTPUT	V\$RMAN_STATUS
V\$ROLLSTAT	V\$ROWCACHE
V\$ROWCACHE_PARENT	V\$ROWCACHE_SUBORDINATE
V\$RSRC_CONSUMER_GROUP	V\$RSRC_CONSUMER_GROUP_CPU_MTH
V\$RSRC_CONS_GROUP_HISTORY	V\$RSRC_PLAN
V\$RSRC_PLAN_CPU_MTH	V\$RSRC_PLAN_HISTORY
V\$RSRC_SESSION_INFO	V\$RULE
V\$RULE_SET	V\$RULE_SETAggregate_STATS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น-ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

V\$\$SCHEDULER_RUNNING_JOBS	V\$\$SEGMENT_STATISTICS
V\$\$SEGSTAT	V\$\$SEGSTAT_NAME
V\$\$SERVICEMETRIC	V\$\$SERVICEMETRIC_HISTORY
V\$\$SERVICES	V\$\$SERVICE_EVENT
V\$\$SERVICE_STATS	V\$\$SERVICE_WAIT_CLASS
V\$\$SERV_MOD_ACT_STATS	V\$\$SESSION
V\$\$SESSION_CONNECT_INFO	V\$\$SESSION_CURSOR_CACHE
V\$\$SESSION_EVENT	V\$\$SESSION_LONGOPS
V\$\$SESSION_OBJECT_CACHE	V\$\$SESSION_WAIT
V\$\$SESSION_WAIT_CLASS	V\$\$SESSION_WAIT_HISTORY
V\$\$SESSMETRIC	V\$\$SESSTAT
V\$\$SESS_IO	V\$\$SESS_TIME_MODEL
V\$\$SES_OPTIMIZER_ENV	V\$\$SGA
V\$\$SGAINFO	V\$\$SGASTAT
V\$\$SGA_CURRENT_RESIZE_OPS	V\$\$SGA_DYNAMIC_COMPONENTS
V\$\$SGA_DYNAMIC_FREE_MEMORY	V\$\$SGA_RESIZE_OPS
V\$\$SGA_TARGET_ADVICE	V\$\$SHARED_POOL_ADVICE
V\$\$SHARED_POOL_RESERVED	V\$\$SHARED_SERVER
V\$\$SHARED_SERVER_MONITOR	V\$\$SORT_SEGMENT
V\$\$SORT_USAGE	V\$\$SPPARAMETER
V\$\$SQL	V\$\$SQLAREA
V\$\$SQLAREA_PLAN_HASH	V\$\$SQLSTATS
V\$\$SQLTEXT	V\$\$SQLTEXT_WITH_NEWLINES
V\$\$SQL_BIND_DATA	V\$\$SQL_BIND_METADATA
V\$\$SQL_CURSOR	V\$\$SQL_JOIN_FILTER
V\$\$SQL_OPTIMIZER_ENV	V\$\$SQL_PLAN
V\$\$SQL_PLAN_STATISTICS	V\$\$SQL_PLAN_STATISTICS_ALL
V\$\$SQL_REDIRECTION	V\$\$SQL_SHARED_CURSOR
V\$\$SQL_SHARED_MEMORY	V\$\$SQL_WORKAREA
V\$\$SQL_WORKAREA_ACTIVE	V\$\$SQL_WORKAREA_HISTOGRAM
V\$\$STANDBY_APPLY_SNAPSHOT	V\$\$STANDBY_LOG
V\$\$STATISTICS_LEVEL	V\$\$STATNAME

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VSSTREAMS_APPLY_COORDINATOR	V\$STREAMS_APPLY_READER
VSSTREAMS_APPLY_SERVER	V\$STREAMS_CAPTURE
VSSTREAMS_POOL_ADVICE	V\$STREAMS_TRANSACTION
VSSUBCACHE	V\$SYSAUX_OCCUPANTS
V\$SYSMETRIC	V\$SYSMETRIC_HISTORY
V\$SYSMETRIC_SUMMARY	V\$SYSSTAT
V\$SYSTEM_CURSOR_CACHE	V\$SYSTEM_EVENT
V\$SYSTEM_PARAMETER	V\$SYSTEM_PARAMETER2
V\$SYSTEM_WAIT_CLASS	V\$SYS_OPTIMIZER_ENV
V\$SYS_TIME_MODEL	V\$TABLESPACE
V\$TEMPFILE	V\$TEMPORARY_LOBS
V\$TEMPSTAT	V\$TEMP_CACHE_TRANSFER
V\$TEMP_EXTENT_MAP	V\$TEMP_EXTENT_POOL
V\$TEMP_HISTOGRAM	V\$TEMP_PING
V\$TEMP_SPACE_HEADER	V\$THREAD
V\$THRESHOLD_TYPES	V\$TIMER
V\$TIMEZONE_FILE	V\$TIMEZONE_NAMES
V\$TRANSACTION	V\$TRANSACTION_ENQUEUE
V\$TRANSPORTABLE_PLATFORM	V\$TSM_SESSIONS
V\$TYPE_SIZE	V\$UNDOSTAT
V\$UNUSABLE_BACKUPFILE_DETAILS	V\$VERSION
V\$VPD_POLICY	V\$WAITCLASSMETRIC
V\$WAITCLASSMETRIC_HISTORY	V\$WAITSTAT
V\$WALLET	V\$XML_AUDIT_TRAIL
V\$_LOCK	V\$_LOCK1
V\$_SEQUENCES	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.
ตารางที่ใช้ในการทดลอง

ตาราง STUDENT



student_id	faculty_id	dept_id	citizen_id	pre_name	gender	th_name
th_surname	en_name	en_surname	semester	username	password	birthday
birth_place	race	nationality	religion	blood	number_static	
group_static	lane_static	road_static	district_static	region_static	province_static	
postalcode_static	telephone_static	number_contact	group_contact	lane_contact	road_contact	
district_contact	region_contact	province_contact	postalcode_contact	telephone_contact		
relatives	alive	studying	salary	exam_pattern	exam1	
habital	status_parent	education_status	fin_school	fin_province	fin_year	fin_gpa
before_year	before_univ	before_yuniv	s_parent	sm_parent	std_status	

student_id	รหัสประจำตัวของนักศึกษา โดยกำหนดเป็น Primary key
faculty_id	รหัสของคณะต่างๆในสถาบัน
dept_id	รหัสของภาควิชาต่างๆในสถาบัน
citizen_id	รหัสประจำตัวประชาชน
pre_name	คำนำหน้าชื่อ
gender	เพศ
th_name	ชื่อภาษาไทย
th_surname	นามสกุลภาษาไทย
en_name	ชื่อภาษาอังกฤษ
en_surname	นามสกุลภาษาอังกฤษ
semester	ปีการศึกษาที่เข้าทำการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

username	ชื่อที่ใช้ในการล็อกอินใช้งานระบบ
password	รหัสที่ใช้ในการล็อกอินใช้งานระบบ
birthday	วัน/เดือน/ปี เกิด
birth_place	สถานที่เกิด
race	เชื้อชาติ
nationality	สัญชาติ
religion	ศาสนา
blood	หมู่เลือด
number_static	บ้านเลขที่ ตามทะเบียนบ้าน
group_static	หมู่ที่ ตามทะเบียนบ้าน
lane_static	ตรอก/ซอย ตามทะเบียนบ้าน
road_static	ถนน ตามทะเบียนบ้าน
district_static	ตำบล/แขวง ตามทะเบียนบ้าน
region_static	อำเภอ/เขต ตามทะเบียนบ้าน
province_static	จังหวัด ตามทะเบียนบ้าน
postalcode_static	รหัสไปรษณีย์ ตามทะเบียนบ้าน
telephone_static	โทรศัพท์ ตามทะเบียนบ้าน
number_contact	บ้านเลขที่ ที่ติดต่อได้
group_contact	หมู่ที่ ที่ติดต่อได้
lane_contact	ตรอก/ซอย ที่ติดต่อได้
road_contact	ถนน ที่ติดต่อได้
district_contact	ตำบล/แขวง ที่ติดต่อได้
region_contact	อำเภอ/เขต ที่ติดต่อได้
province_contact	จังหวัด ที่ติดต่อได้
postalcode_contact	รหัสไปรษณีย์ ที่ติดต่อได้
telephone_contact	โทรศัพท์ ที่ติดต่อได้
relatives	นักศึกษาเป็นบุตรคนที่ในจำนวนพี่น้องบิดา-มารดาเดียวกัน
alive	จำนวนพี่น้องบิดา-มารดาที่ยังมีชีวิตอยู่
studying	จำนวนของพี่น้องที่อยู่ในระหว่างการศึกษา
salary	รายได้จากการทำงาน
exam_pattern	รูปแบบการสอบผ่านทบวงมหาวิทยาลัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

exam1	การดำเนินการจัดสอบ/คัดเลือกหลักสูตร4-5ปี
habital	ปัจจุบันอาศัยอยู่กับ
status_parent	สถานภาพสมรสของบิดา-มารดา
education_status	สถานะการจบการศึกษา (ม.6 – สอบเทียบ-ปวส)
fin_school	สำเร็จการศึกษามาจากโรงเรียน
fin_province	สำเร็จการศึกษามาจากจังหวัด
fin_year	สำเร็จการศึกษาในปีการศึกษา
fin_gpa	คะแนนเฉลี่ยสะสมที่สำเร็จการศึกษา
before_year	ปีที่เคยเป็นนักศึกษาของสถาบันฯมาก่อน
before_univ	ชื่อมหาวิทยาลัย/สถาบันฯที่เคยศึกษามาก่อน
before_yuniv	ปีการศึกษาของมหาวิทยาลัย/สถาบันฯที่เคยทำการศึกษามาก่อน
s_parent	ผู้ที่เป็นผู้ปกครอง
sm_parent	ผู้อุปการะทางการเงิน
std_status	สถานะความเป็นนักศึกษา

สร้างตาราง student โดยใช้ ไฟล์ create_student.sql

เพิ่มข้อมูลลงในตารางโดยใช้ ไฟล์ insert_student.sql

โปรแกรม gen_insertrow_student.py เป็นโปรแกรมที่ใช้สร้างไฟล์ insert_student.sql

ตาราง BOOK



bookid	bnamethai	bnameeng	advisor1	Advisor2
Advisor3	student1	Studentid1	Email1	Student2
studentid2	Email2	Student3	Studentid3	Email3
Student4	Studentid4	Email4	bookyear	

bookid รหัสของหนังสือ โดยกำหนดเป็น Primary key

bnamethai ชื่อหนังสือโครงการภาษาไทย

bnameeng ชื่อหนังสือโครงการภาษาอังกฤษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

advisor1 ชื่ออาจารย์ที่ปรึกษาโครงการคนที่ 1
 advisor2 ชื่ออาจารย์ที่ปรึกษาโครงการคนที่ 2
 advisor3 ชื่ออาจารย์ที่ปรึกษาโครงการคนที่ 3
 student1 ชื่อนักศึกษาเจ้าของโครงการคนที่ 1
 studentid1 รหัสของนักศึกษาเจ้าของโครงการคนที่ 1
 email1 อีเมลของนักศึกษาเจ้าของโครงการคนที่ 1
 student2 ชื่อนักศึกษาเจ้าของโครงการคนที่ 2
 studentid2 รหัสของนักศึกษาเจ้าของโครงการคนที่ 2
 email2 อีเมลของนักศึกษาเจ้าของโครงการคนที่ 2
 student3 ชื่อนักศึกษาเจ้าของโครงการคนที่ 3
 studentid3 รหัสของนักศึกษาเจ้าของโครงการคนที่ 3
 email3 อีเมลของนักศึกษาเจ้าของโครงการคนที่ 3
 student4 ชื่อนักศึกษาเจ้าของโครงการคนที่ 4
 studentid4 รหัสของนักศึกษาเจ้าของโครงการคนที่ 4
 email4 อีเมลของนักศึกษาเจ้าของโครงการคนที่ 4
 bookyear ปีการศึกษาที่จัดทำโครงการ

สร้างตาราง book โดยใช้ ไฟล์ create_book.sql

เพิ่มข้อมูลลงในตารางโดยใช้ ไฟล์ insert_book.sql

โปรแกรม gen_insertrow_book.py เป็นโปรแกรมที่ใช้สร้างไฟล์ insert_book.sql

ตาราง BOOKBORROW

	←————→	F.K.	F.K.	
borrow_id	student_id	bookid	dborrow	bstatus

borrow_id ลำดับของการยืมหนังสือ โดยกำหนดเป็น Primary key

student_id รหัสของนักศึกษา

bookid รหัสของหนังสือ

dborrow วันที่ทำการยืมหนังสือ

bstatus สถานะของหนังสือว่าถูกยืมไปหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สร้างตาราง borrow โดยใช้ ไฟล์ create_borrow.sql

เพิ่มข้อมูลลงในตารางโดยใช้ ไฟล์ insert_borrow.sql

โปรแกรม gen_insertrow_borrow.py เป็นโปรแกรมที่ใช้สร้างไฟล์ insert_borrow.sql



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้