บทที่ 3
วิธีดำเนินการวิจัย

งานวิจัยนี้เป็นการนำเสนอ การเปรียบเทียบสมรรถนะการทำงานระหว่าง ระบบจัดการฐานข้อมูลออราเคิล เวอร์ชั่น 9i และเวอร์ชั่น 10g ในแง่ของสมรรถนะการทำงานของระบบจัดการฐานข้อมูลที่ใช้บนแพลตฟอร์มยูนิกซ์ ในการศึกษาการปรับเปลี่ยนเวอร์การทำงานของระบบจัดการฐานข้อมูลออราเคิลนั้น มีหัวข้อที่เกี่ยวข้องสำหรับในบทที่ 3 คือ การเตรียมสภาพแวดล้อมและองค์ประกอบเพื่อใช้ในการวิจัย ซึ่งจะอธิบายถึงการเตรียม ฮาร์ดแวร์ ซอฟต์แวร์ของระบบจัดการฐานข้อมูลออราเคิล โปรแกรมที่ใช้ในการทดสอบ ซึ่งเป็นการใช้งานร่วมกันระหว่างเชลล์สคริปต์และเอสคิวแอลพลัส รายละเอียดยูนิกซ์ ซอฟต์แวร์ที่ใช้สำหรับวัดสมรรถนะ ที่เป็นเครื่องมือของบีเอ็มซีซอฟต์แวร์ ชื่อพาโทรล และข้อมูลที่ใช้ในการทดสอบกรณีศึกษา เมื่อเตรียมความพร้อมของอุปกรณ์และโปรแกรมเรียบร้อยแล้ว ขั้นตอนต่อไป คือหลักการวิเคราะห์ข้อมูลซึ่งอธิบายถึงเหตุการณ์ที่ใช้เป็นกรณีศึกษา เกณฑ์ที่ใช้ในการเปรียบเทียบสมรรถนะ และรายละเอียดของการเก็บรวบรวมข้อมูล ซึ่งรายละเอียดต่างๆ ที่ได้กล่าวมาแล้วนั้น มีดังต่อไปนี้

3.1 การเตรียมสภาพแวดล้อม และองค์ประกอบเพื่อใช้ในการวิจัย

3.1.1 ฮาร์ดแวร์

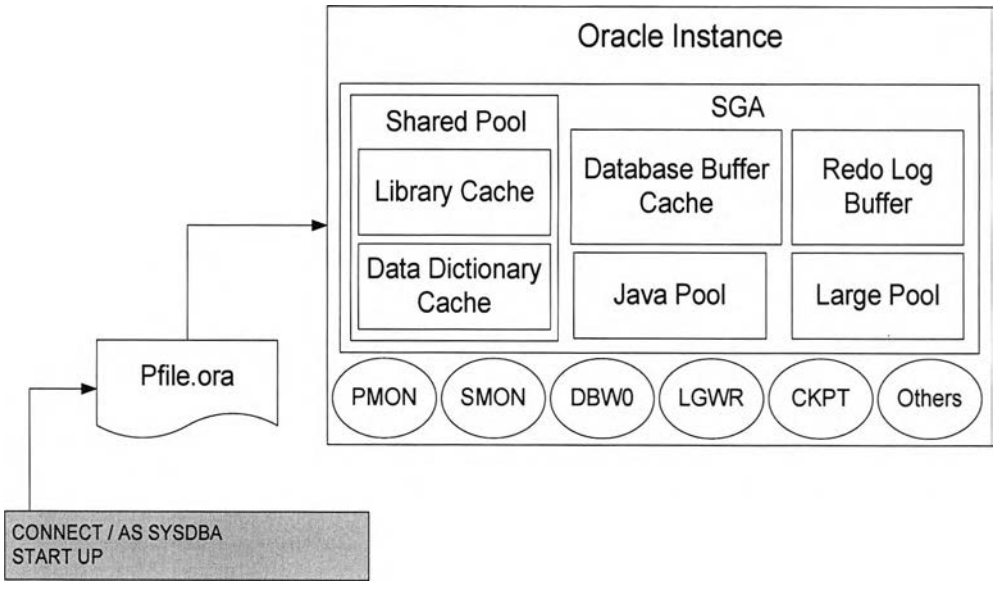
ฮาร์ดแวร์เป็นแพลตฟอร์มยูนิกซ์จำนวน 1 เครื่อง มีรายละเอียดแสดงดังตารางที่ 3.1

ตารางที่ 3.1 คุณสมบัติของฮาร์ดแวร์

คุณสมบัติ	เซิฟเวอร์หลัก
ระบบปฏิบัติการ	Sun solaris 5.8
CPU	CPU 900 MHz , 2 CPU
หน่วยความจำ	RAM 2048 MB
หน่วยความจำสำรอง	HARDDISK 80 GB
IP Address	172.16.15.199
MODEL	SunBlade 2000

3.1.2 ซอฟต์แวร์ของระบบจัดการฐานข้อมูลออราเคิล

ซอฟต์แวร์ที่ใช้เป็นระบบฐานข้อมูลออราเคิลเวอร์ชั่น 9.2.0.1 ออราเคิลเวิร์ชั่น 10.1.0.2 และออราเคิลเวอร์ชั่น 10.2.0.1 ในระบบการจัดการฐานข้อมูลออราเคิลนั้นต้องมีการกำหนดพารามิเตอร์ที่จำเป็นตามค่ามาตรฐานที่ทางระบบการติดตั้งได้กำหนดไว้ พารามิเตอร์เหล่านี้จะถูกเรียกทำงานทุกครั้งเมื่อมีการเริ่มต้นการใช้งานอินสแตนส์ โดยพารามิเตอร์เหล่านี้จะเก็บอยู่ในรูปของพารามิเตอร์ไฟล์ หรือพีไฟล์ (PFILE) ซึ่งเป็นข้อมูลประเภทเท็กซ์ไฟล์ พีไฟล์นี้ในขณะที่มีการใช้งานของอินสแตนส์นั้น จะอยู่ในสถานะอ่านได้อย่างเดียว (Read Only) หากมีการแก้ไขค่าของพารามิเตอร์ที่กำหนดในพีไฟล์จะต้องทำการออกจากการใช้งานอินสแตนส์ และเข้ามาใช้งานใหม่เพื่อให้มีการเรียกใช้ค่าพารามิเตอร์ที่มีการเปลี่ยนแปลงนั้นเสมอ และจะเป็นค่านั้นตลอดไปจนกว่าจะมีการแก้ไขและเริ่มต้นใช้งานอินสแตนส์นั้นใหม่ พีไฟล์นี้จะใช้ชื่อในการเก็บพารามิเตอร์ว่า initSID.ora ลักษณะความสัมพันธ์ของพีไฟล์กับออราเคิลอินสแตนส์แสดงดังรูปที่ 3.1



รูปที่ 3.1 แสดงเอกสารการกำหนดค่าเริ่มต้น หรือ พีไฟล์ในระบบจัดการฐานข้อมูลออราเคิล

ไฟล์ initSID.ora เป็นพารามิเตอร์ไฟล์เก็บตามชื่อแต่ละระบบฐานข้อมูลออราเคิล ภายใต้ตัวแปรที่ชื่อ SID ตัวอย่างเช่น ชื่อไฟล์ initORACL1.ora แสดงว่าฐานข้อมูลนี้ชื่อ ORACL1 เป็นต้น ภายในพารามิเตอร์ไฟล์นี้ ประกอบด้วยพารามิเตอร์ต่างๆ ดังแสดงตัวอย่างในตารางที่ 3.2 และรูปที่ 3.2 แสดงพีไฟล์ที่ใช้ในงานวิจัยของระบบฐานข้อมูลออราเคิลเวอร์ชั่น 9.2.0.1 ที่ได้มาจากการติดตั้งระบบจัดการฐานข้อมูลออราเคิลเวอร์ชั่น 9.2.0.1 แบบมาตรฐาน

ตารางที่ 3.2 ตัวอย่างพารามิเตอร์ที่มีการกำหนดในพีไฟล์

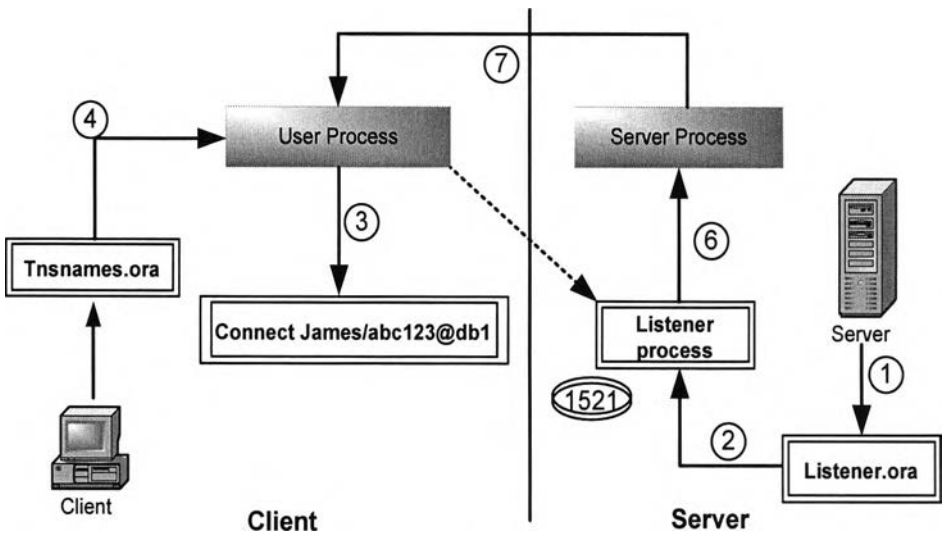
พารามิเตอร์	ประเภทของพารามิเตอร์	ความหมายของพารามิเตอร์
BACKGROUND_DUMP_DEST	String	ที่อยู่สำหรับเก็บล็อกไฟล์ ของกระบวนการทำงานในระบบฐานข้อมูลออราเคิล กรณีที่เกิดข้อผิดผลาดในการทำงาน
COMPATIBLE	String	เก็บออราเคิลเวิร์ชั่นที่ ออราเคิลอินสแตนส์นี้สามารถใช้งานด้วยกันได้
CONTROL_FILES	String	ระบุที่อยู่และชื่อของคอนโทรลไฟล์ ของออราเคิลอินสแตนส์
DB_BLOCK_SIZE	Integer	ขนาดของบล็อคของออราเคิลดาต้าเบส
DB_FILE_MULITBLOCK_READ_COUNT	Integer	จำนวนบล็อคของข้อมูลที่อ่านมาจากไฟล์ข้อมูลในแต่ละครั้ง
DB_NAME	String	เก็บชื่อของระบบฐานข้อมูล เป็นตัวอักษรไม่เกิน 8 ตัวอักษร
DB_FILES	String	จำนวนไฟล์ที่มีได้สูงสุดในฐานข้อมูล
PARALLEL_MAX_SERVERS	Integer	จำนวนโปรเซสลูงสุดที่สร้างขึ้นได้ในฐานข้อมูล
OPER_CURROR		จำนวนฐานข้อมูลสูงสุดของฐานข้อมูลที่สามารถเปิดใช้งานได้
SHARED_POOL_SIZE	Integer	จำนวนพื้นที่ในหน่วยความจำในเอสจีเอที่จองไว้ใช้กับไลเบอร์แคช และดาต้าดิกชั่นนารีแคช ในฐานข้อมูลมีหน่วยเป็นไบต์
SORT_AREA_SIZE	Integer	จำนวนไบต์ของหน่วยความจำที่ใช้ในการเรียงลำดับข้อมูลสำหรับการทำงานของยูสเซอร์โปรเซส
USER_DUMP_DEST	String	พาธที่เก็บล็อกไฟล์ของยูสเซอร์โปรเซส กรณีเกิดข้อผิดพลาดในการทำงานของยูสเซอร์โปรเซส

```
##########################################################################
# Copyright (c) 1991, 2001, 2002 by Oracle Corporation
##########################################################################
# Cache and I/O
db_block_size=8192
db_cache_size=33554432
db_file_multiblock_read_count=16
 # Cursors and Library Cache
open_cursors=300
 # Database Identification
db_domain=""
db_name=ORA92
 # Diagnostics and Statistics
background_dump_dest=/data/oracle92/OraHome1/admin/ORA92/bdump
core_dump_dest=/data/oracle92/OraHome1/admin/ORA92/cdump
timed_statistics=TRUE
user_dump_dest=/data/oracle92/OraHome1/admin/ORA92/udump
 # File Configuration
control_files=("/data/oracle92/OraHome1/oradata/ORA92/control01.ctl",
"/data/oracle92/OraHome1/oradata/ORA92/control02.ctl",
"/data/oracle92/OraHome1/oradata/ORA92/control03.ctl")
# Instance Identification
instance_name=ORA92
# Job Queues
job_queue_processes=10
 # MTS
dispatchers="(PROTOCOL=TCP) (SERVICE=ORA92XDB)"
 # Miscellaneous
aq_tm_processes=1
compatible=9.2.0.0.0
 # Optimizer
hash_join_enabled=TRUE
query_rewrite_enabled=FALSE
# Pools
java_pool_size=117440512
large_pool_size=16777216
shared_pool_size=117440512
 # Processes and Sessions
processes=150
# Redo Log and Recovery
fast_start_mttr_target=300
# Security and Auditing
remote_login_passwordfile=EXCLUSIVE
# Sort, Hash Joins, Bitmap Indexes
pga_aggregate_target=25165824
sort_area_size=524288
# System Managed Undo and Rollback Segments
undo_management=AUTO
undo_retention=10800
undo_tablespace=UNDOTBS1
```

รูปที่ 3.2 แสดงไฟล์ *initORA92.ora* ของระบบฐานข้อมูลออราเคิลเวอร์ชัน 9.2.0.1

ขั้นตอนที่ใช้ติดต่อเข้ามายังฐานข้อมูล การติดต่อเข้ามายังฐานข้อมูล ออราเคิลที่เครื่องเซิร์ฟเวอร์จากไคลเอนท์นั้น สามารถทำได้โดยใช้หลักการเดียวกันกับการสร้างการติดต่อแบบไคลเอนท์เซิร์ฟเวอร์ โดยมียูสเซอร์โปรเซสที่ฝั่งไคลเอนท์ร้องขอบริการไปยัง เซิร์ฟเวอร์โปรเซส ที่ฝั่งเซิร์ฟเวอร์ จากนั้นทั้ง 2 ฝั่งคือ ไคลเอนท์และเซิร์ฟเวอร์ จะติดต่อกันด้วยโปรโตคอลตามที่กำหนด ซึ่งทั้ง ยูสเซอร์โปรเซสและเซิร์ฟเวอร์โปรเซสจะต้องเป็นโปรโตคอล ทีซีพี ไอพี (TCP/IP) เป็นต้น

ในฝั่งเซิร์ฟเวอร์ที่เป็นฐานข้อมูลออราเคิลนั้นจะต้องเปิดบริการที่เรียกว่า ลีซเทนเนอร์โปรเซส (Listener Process) เพื่อรองรับการติดต่อจากไคลเอนท์เข้ามายังฐานข้อมูล ลีซเทนเนอร์นั้นจะเป็นส่วนที่จับคู่ยูสเซอร์โปรเซส จากไคลเอนท์ให้กับเซิร์ฟเวอร์โปรเซส ที่ฝั่งเซิร์ฟเวอร์โดยจะติดต่อจากช่องทางที่กำหนด การกำหนดพอร์ตและชื่อของลีซเทนเนอร์นั้น สามารถกำหนดได้ที่ไฟล์ Listener.ora ซึ่งจะเก็บค่าการติดตั้งสำหรับใช้ในฐานข้อมูล หากไม่มีการติดตั้งลีซเทนเนอร์ไว้ที่เซิร์ฟเวอร์ จะทำให้ไคลเอนท์ไม่สามารถที่ติดต่อเข้ามายังฐานข้อมูลได้ การติดต่อเข้ามายังฐานข้อมูลออราเคิลนั้น แสดงดังรูปที่ 3.3



รูปที่ 3.3 แสดงการสร้างการติดต่อกับฐานข้อมูล

จากรูปที่ 3.3 สามารถอธิบายขั้นตอนในการติดต่อจากไคลเอนท์เข้าไปยังฐานข้อมูลออราเคิล ได้ดังนี้

1. ที่ฝั่งเซิร์ฟเวอร์เริ่มต้นสร้างลีซเทนเนอร์โปรเซส ขึ้นมารองรับการทำงานของยูสเซอร์โปรเซส เพื่อจับคู่กับเซิร์ฟเวอร์โปรเซสโดยเปิดอ่านค่าติดตั้งของลีซเทนเนอร์ จากไฟล์ Listener.ora

2. สร้างลีซเทนเนอร์โปรเซสตามค่าที่กำหนดในไฟล์ Listener.ora โดยจะมีชื่อและมอนิเตอร์การติดต่อตามพอร์ตที่กำหนด จากรูปที่ 3.3 จะมอนิเตอร์การติดต่อที่เข้ามาที่พอร์ต 1521 เท่านั้น

3. ยูสเซอร์โปรเซส ได้รับคำสั่งให้ติดต่อฐานข้อมูลมาจากยูสเซอร์ โดยคำสั่งที่ส่งมา คือ *Connect James/abc123@db1* แสดงว่าต้องการติดต่อไปยังบริการที่ชื่อว่า db1 โดยติดต่อเป็นชื่อผู้ใช้ James และรหัสผ่าน คือ abc123

4. ยูสเซอร์โปรเซส จะทำการตรวจสอบดูข้อมูลจากไฟล์ Tnsnames.ora ที่ฝั่งไคลเอนท์ว่า ชื่อของบริการที่ส่งมานั้น เป็นฐานข้อมูลชื่อว่าอะไร อยู่บนเครื่องเซิร์ฟเวอร์ไหน และต้องติดต่อไปยังพอร์ตอะไร

5. สร้างการติดต่อไปยังเซิร์ฟเวอร์ที่อ่านค่ามาได้จากไฟล์ Tnsnames.ora ตามพอร์ตที่กำหนด จากตัวอย่างตามรูป 3.3 คือพอร์ตที่ 1521

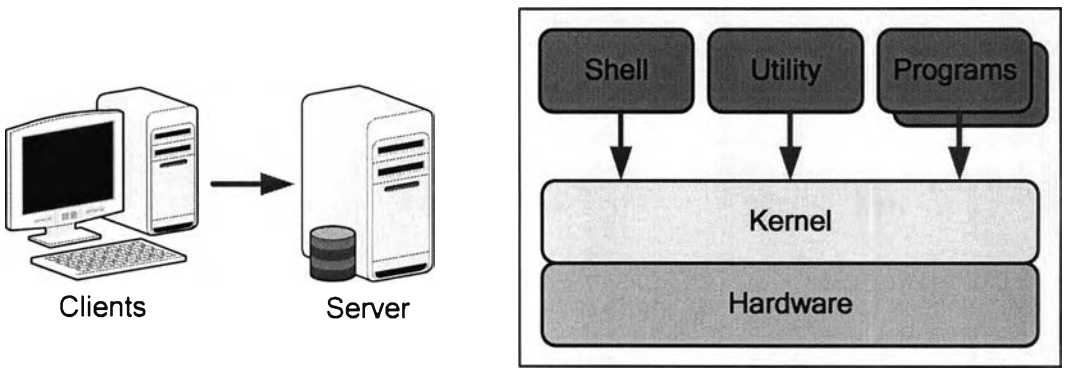
6. ลีซเทนเนอร์โปรเซสรับทราบการติดต่อจากพอร์ต 1521 และตรวจสอบดูว่าฐานข้อมูลที่ต้องการติดต่อนั้นเป็นฐานข้อมูลเดียวกันกับที่ลีซเทนเนอร์ต้องสร้างการติดต่อหรือไม่ เมื่อผ่านตรวจสอบว่าค่าของฐานข้อมูลตรงกันแล้ว จะจับคู่ยูสเซอร์โปรเซสนั้นให้กับเซิร์ฟเวอร์โปรเซสในฐานข้อมูลเพื่อทำงานตามที่กำหนด

7. เมื่อเซิร์ฟเวอร์โปรเซสทำงานเสร็จ แล้วจะส่งผลลัพธ์กลับคืนให้กับยูสเซอร์โปรเซส

3.1.3 **สำหรับโปรแกรมที่ใช้ในการทดสอบนั้น เป็นการใช้งานร่วมกันระหว่างโปรแกรมเชลล์ หรือ เชลล์สคริปต์** (Shell script) **บนระบบปฏิบัติการยูนิกซ์ และโปรแกรมเอสคิวแอลพลัส** (SQL *Plus)

1) **ยูนิกซ์** คือ ชื่อของระบบปฏิบัติการ (Operating System : O.S) ระบบหนึ่งที่ได้รับการออกแบบพัฒนาเพื่อใช้งานกับระบบคอมพิวเตอร์ที่สามารถใช้งานพร้อมกันได้หลายๆ ผู้ใช้บริการ (Multiuser) และทำงานแบบที่มีการตอบโต้ระหว่างระบบกับผู้ใช้งานทันที

องค์ประกอบโครงสร้างของยูนิกซ์ แบ่งออกได้เป็น 3 ส่วนตามที่แสดงในรูปที่ 3.4 คือ ส่วนที่แนบชิดติดกับฮาร์ดแวร์เรียกว่า เคอร์เนล (Kernel) ซึ่งทำหน้าที่ดูแลการจัดอันดับการทำงาน และควบคุมการใช้ทรัพยากรต่างๆในระบบคอมพิวเตอร์ ส่วนต่อมาเป็นส่วนที่เรียกว่าเชลล์ซึ่งเปรียบเสมือนสื่อกลางระหว่างยูนิกซ์กับผู้ใช้ ทำหน้าที่ในการแปลคำสั่งตามที่ผู้ใช้ และในส่วนนอกสุดนั้น คือส่วนที่เป็นยูทิลิตี้ (Utility) และโปรแกรมประยุกต์ต่างๆ ซึ่งทำให้ยูนิกซ์มีความสามารถในฟังก์ชันการทำงานมากขึ้น



รูปที่ 3.4 แสดงโครงสร้างของยูนิกซ์

(1) เคอร์เนล ส่วนของเคอร์เนลถือเป็นหัวใจของยูนิกซ์ เพราะเป็นส่วนที่ทำการควบคุมการทำงานของฮาร์ดแวร์ เช่น ควบคุมการสวิตซ์โปรเซสเซอร์ไปมาระหว่างงานหลายๆ งาน และควบคุมการทำงานของจานแม่เหล็ก เป็นต้น เนื่องจากขอบเขตงานที่เคอร์เนลดูแลอยู่เป็นสิ่งที่ต้องเรียกให้กระทำอยู่บ่อยๆ (คือหลายครั้งใน 1 วินาที) จึงมีความจำเป็นอย่างยิ่งที่ต้องให้ส่วนที่เป็นเคอร์เนลนี้อยู่ในหน่วยความจำตลอดเวลา การออกแบบยูนิกซ์จึงได้ยึดหลักที่ว่า ให้เคอร์เนลมีขนาดเล็กที่สุดเท่าที่จะเป็นไปได้ เพื่อที่จะได้ไม่ใช้เนื้อที่ในหน่วยความจำหลักเกินความจำเป็น

(2) เชลล์ อย่างที่กล่าวไปแล้ว เชลล์ทำตัวเสมือนเป็นสื่อกลางระหว่างผู้ใช้กับเคอร์เนล ดังนั้นโดยชื่อของเชลล์อาจสร้างภาพพจน์ได้ว่า เชลล์ คือ เปลือกที่ห่อหุ้มเคอร์เนลเอาไว้ หน้าที่สำคัญของเชลล์ คือเป็นล่ามแปลคำสั่งต่างๆที่ผู้ใช้ต้องการให้เคอร์เนลเข้าใจ เพื่อจะได้ประกอบการควบคุมฮาร์ดแวร์ให้ปฏิบัติตามคำสั่งนั้นๆ นอกจากนี้แล้วเชลล์ยังเป็นภาษาโปรแกรมอีกด้วยโดยผู้ใช้สามารถเขียนโปรแกรมขึ้นมาโดยใช้คำสั่งเชลล์

(3) ยูทิลิตี้และโปรแกรมประยุกต์ โดยเฉลี่ยแล้วระบบยูนิกซ์จะประกอบขึ้นด้วยยูทิลิตี้ต่างๆ กว่า 200 ฟังก์ชัน ส่วนของยูทิลิตี้และซอฟต์แวร์ประยุกต์นี้จะเก็บไว้ในหน่วยความจำสำรอง และจะถูกนำเข้ามายังหน่วยความจำหลักก็เฉพาะเมื่อมีการเรียกใช้ยูทิลิตี้ เป็นส่วนที่ใช้ทำงานทั่วๆ ไปในขณะที่โปรแกรมประยุกต์จะทำงานเฉพาะกิจ เช่น โปรแกรมประยุกต์สำหรับระบบบัญชี โปรแกรมเอดิเตอร์ (Editor) เป็นต้น ส่วนตัวอย่างของยูทิลิตี้ก็ได้แก่ ยูทิลิตี้สำหรับแสดงวัน เวลา วันที่ยูทิลิตี้ในการเรียงลำดับข้อความ เป็นต้น นอกจากนี้ยูนิกซ์ยังประกอบด้วยเครื่องมือซอฟต์แวร์ต่างๆ เช่นเครื่องมือช่วยพัฒนาโปรแกรม เป็นต้น

จากที่ได้อธิบายในข้างต้นจึงอาจกล่าวได้ว่า เชลล์เป็นได้ทั้งตัวแปลคำสั่ง และภาษาโปรแกรม เชลล์สคริปต์ก็คือโปรแกรมเชลล์ อันได้แก่ การนำเอาคำสั่งเชลล์มาประกอบกับมากกว่า 1 คำสั่ง รวมถึงการนำประโยคเงื่อนไข ลูป และอื่นๆ รวมทั้งการใช้ตัวแปรมาประกอบกันเป็นโปรแกรม ส่วนในแง่ของการเปรียบเทียบระหว่างโปรแกรมเชลล์ กับภาษาโปรแกรมทั่วๆ ไป สิ่งแรกก็คือ เชลล์เป็นภาษาแบบที่มีการตอบโต้ระหว่างระบบกับผู้ใช้งานทันที ในขณะที่ภาษาอื่นๆ ส่วนมากจะเป็นคอมไพเลอร์ นั่นก็หมายความว่าเชลล์จะต้องทำการแปลโปรแกรมทุกครั้งที่รันโปรแกรม ทำให้โปรแกรมเชลล์นี้ทำงานได้ช้ากว่าโปรแกรมที่เขียนด้วยภาษาอื่น แต่จุดเด่นของภาษาเชลล์ก็คือความง่ายในการเขียนโปรแกรม โดยเฉพาะกับลักษณะงานที่มีคำสั่งเชลล์ใช้อยู่แล้ว นอกจากนี้ตัวแปรในโปรแกรมเชลล์จะเป็นในลักษณะที่ไม่ระบุชนิด (weak type) เช่น การอ่านข้อมูลจากเทอร์มินัลไปเก็บไว้ในตัวแปร สามารถใช้ตัวแปรเดียวกันได้ ไม่ว่าข้อมูลที่ผู้ใช้พิมพ์นั้นจะเป็นตัวเลข หรือข้อความ บนระบบยูนิกซ์ก็มีคำสั่งที่เขียนขึ้นด้วยเชลล์ เช่น กระบวนการในการปิดระบบ เป็นต้น

2) **โปรแกรมเอสคิวแอลพลัส** เป็นโปรแกรมมาตรฐานของออราเคิล ที่ใช้ในการติดต่อจากไคลเอนท์ไปยังฐานข้อมูล โดยโปรแกรมนี้จะเป็นเครื่องมือช่วยให้ผู้ใช้สามารถใช้คำสั่งเอสคิวแอล และเซตของคำสั่งเอสคิวแอล ผ่านภาษา พีแอลเอสคิวแอล (PL/SQL) ได้ ซึ่งลักษณะการทำงานของโปรแกรมเอสคิวแอลพลัสจะเป็นเท็กซ์โหมด และการทำงานนี้จะมีเนื้อที่สำรอง (Buffer) ที่จัดเก็บคำสั่งที่ส่งไปทำงานกับฐานข้อมูลเอาไว้ ผู้ใช้จึงสามารถที่จะเรียกดูคำสั่งสุดท้ายซึ่งเก็บไว้ในเนื้อที่สำรองมาแก้ไข และใช้งานใหม่ได้ คำสั่งที่สามารถเรียกใช้ภายในโปรแกรมเอสคิวแอลพลัส ได้จะแบ่งออกเป็น 2 แบบคือ

(1) แบบที่ 1 คำสั่งเอสคิวแอลสเตทเม็นท (SQL Statement) เป็นลักษณะคำสั่งเอสคิวแอล ที่ใช้ทำงานกับฐานข้อมูลของออราเคิล ซึ่งจะอ้างอิงมาจากภาษาเอสคิวแอลมาตรฐานที่ใช้งานอยู่ทั่วไป เช่น *Select Insert Delete* เป็นต้น นอกจากนี้สามารถใช้ พีแอลเอสคิวแอล ซึ่งเป็นฟังก์ชันการทำงานเพิ่มเติมของออราเคิล ในเอสคิวแอลพลัสได้อีกด้วย โดยตัวอย่างของพีแอลเอสคิวแอล เช่น *Begin* หรือ *Declare* เป็นต้น

(2) แบบที่ 2 คำสั่ง เอสคิวแอลพลัส (SQL *Plus) เป็นลักษณะคำสั่งเฉพาะของเอสคิวแอลพลัส ที่เอาไว้ใช้จัดการกับการทำงานภายในโปรแกรม เอสคิวแอลพลัสเองไม่ได้เอาไว้ใช้จัดการกับข้อมูลในฐานข้อมูล เช่น *Append Change Del Input* เป็นต้น ข้อดีอีกอย่างของคำสั่งเอสคิวแอลพลัส ก็คือ ผู้ใช้ไม่จำเป็นต้องพิมพ์คำสั่งครบทุกคำ แต่สามารถที่จะใช้คำย่อเพื่อแทนแต่ละคำสั่งได้ เช่น a แทน *append* เป็นต้น

I 28659 98

ในการสร้างโปรแกรมร่วมกันระหว่างเชลล์สคริปต์ และพีแอลเอสคิวแอลนั้น เป็นลักษณะของการทำงานแบบเรียงลำดับ คือ เป็นการนำชุดคำสั่งของเชลล์มาเรียกกันในชุดโปรแกรม และในชุดโปรแกรมนั้นสามารถที่จะระบุ ชื่อโปรแกรมของพีแอลเอสคิวแอลที่ต้องการให้ทำงานภายใต้ชุดคำสั่งของเชลล์ ดังตัวอย่างเชลล์สคริปต์นี้ชื่อ step1_create_table.sh มีสกุลเป็น .sh แสดงให้เห็นว่าเป็นชนิดของไฟล์เชลล์สคริปต์ ประกอบด้วย 2 คำสั่ง คือ

บรรทัดที่ 1 **DT=`date +%Y%m%d_%H:%M` ;export DT** เป็นการให้แสดง วัน เวลา

บรรทัดที่ 2 **sqlplus -s paricm11/duean2515@ora92 @step1_create_tab1.sql > step1_create_tab1_$DT.log &** เป็นการเรียกโปรแกรมพีเอลเอสคิวแอลชื่อ step1_create_tab1.sql ขึ้นมาทำงาน และให้เขียนผลของการทำงานในไฟล์ที่ชื่อ step1_create_tab1_$DT.log

3.1.4 ซอฟต์แวร์ที่ใช้สำหรับวัดสมรรถนะ ของ บีเอ็มซี ซอฟต์แวร์ ชื่อพาโทรล

พาโทรล (PATROL®) เป็นเครื่องมือของบริษัท บีเอ็มซี ซอร์ฟแวร์ (BMC Software) ในการจัดการสำหรับเครื่องคอมพิวเตอร์เพื่อใช้สำหรับการตรวจสอบและเก็บข้อมูล เพื่อช่วยให้ผู้ตรวจสอบระบบสามารถตรวจวัดการทำงานและประสิทธิภาพของเครื่องคอมพิวเตอร์ใน แอพพลิเคชั่นหรือระบบปฏิบัติการต่างๆ ในงานวิจัยนี้เป็นพาโทรลที่ใช้สำหรับการตรวจสอบระบบปฏิบัติการบนยูนิกซ์ พาโทรลสามารถตรวจสอบและเก็บข้อมูล ณ ปัจจุบัน รวมถึงทำการเก็บข้อมูลย้อนหลังผ่านการทำงานที่มีส่วนประกอบดังต่อไปนี้

1) **พาโทรลเอเจนท์** มีหน้าที่เก็บข้อมูลตามชุดคำสั่ง (Knowledge Module) ที่ถูกกำหนดไว้ในการกำหนดคุณสมบัติของเอเจนท์ (Agent Configuration) โดย พาโทรลเอเจนท์มีส่วนประกอบต่างๆดังต่อไปนี้

(1) ตัวโปรแกรมการทำงานของพาโทรล (PatrolAgent) ที่มีการอ้างอิงถึงการทำงานตามชุดคำสั่ง โดยจะทำการปฏิบัติการตรวจสอบและเก็บข้อมูลตามรอบเวลาที่กำหนดในการกำหนดคุณสมบัติของพาโทรลเอเจนท์

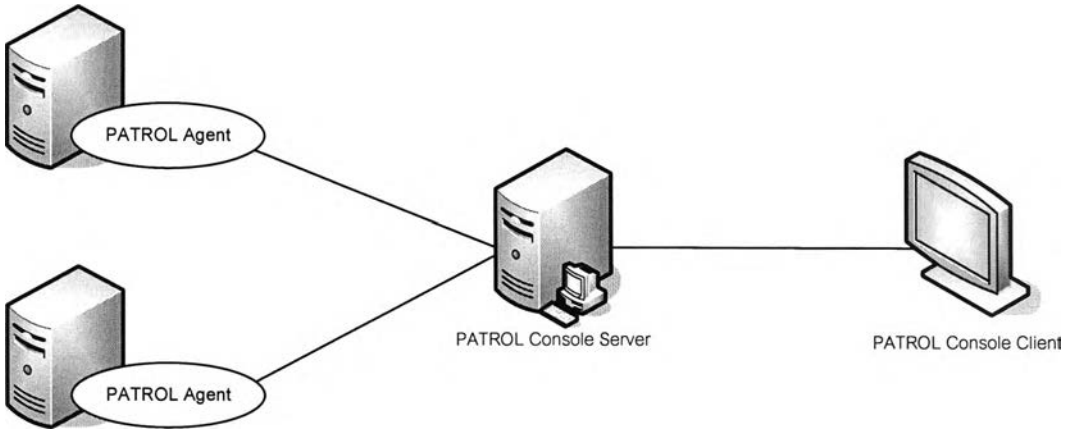
(2) ชุดคำสั่ง (Knowledge Module) มีการแบ่งออกเป็นชุดคำสั่งย่อยสำหรับแอพพลิเคชั่นหรือระบบปฏิบัติการต่างๆ เช่น ชุดคำสั่งสำหรับระบบปฏิบัติการยูนิกซ์ หรือชุดคำสั่งสำหรับระบบฐานข้อมูลไซเบส ซึ่งในแต่ละชุดคำสั่งใหญ่ จะมีการแบ่งชุดคำสั่งย่อยไปตามส่วนของการตรวจสอบ เช่น ชุดคำสั่งในการตรวจสอบการทำงานของซีพียู เป็นต้น (เป็นไปตามการเลือกติดตั้ง)

(3) การกำหนดคุณสมบัติของพาโทรลเอเจนท์ เช่น การเลือกใช้ชุดคำสั่งย่อยในการตรวจสอบและเก็บข้อมูล ระยะเวลาในการเก็บข้อมูลย้อนหลังในอดีต ข้อมูลของเครื่องพาโทรลคอนโซลที่ใช้ในการเชื่อมต่อ เป็นต้น

(4) ข้อมูลจากการตรวจสอบและเก็บข้อมูลปัจจุบันย้อนหลังไปในอดีต ซึ่งมีการเก็บเป็นข้อมูลในรูปแบบไบนารี่ (เป็นไปตามการกำหนดคุณสมบัติ)

2) **พาโทรลคอนโซลเซิร์ฟเวอร์** มีหน้าที่เชื่อมต่อกับพาโทรลเอเจนท์ เพื่อทำการดึงข้อมูลจากพาโทรลเอเจนท์มาแสดงผลบนหน้าจอพาโทรลคอนโซลไคล์แอนท์ และเป็นหน้าจอในการกำหนดคุณสมบัติของพาโทรลเอเจนท์ อีกทั้งยังมีหน้าที่ในการกำหนดคุณสมบัติการใช้งานผ่านพาโทรลคอนโซลไคลแอนท์ เช่น ยูสเซอร์ที่มีสิทธิในการใช้งาน สิทธิในการเรียกใช้ สิทธิในการกำหนดคุณสมบัติต่างๆจากพาโทรลคอนโซลไคลแอนท์

3) **พาโทรลคอนโซลไคลแอนท์** มีหน้าที่เป็นหน้าจอแสดงผลจากข้อมูลของพาโทรลเอเจนท์ออกมาในรูปแบบกราฟฟิคและตัวอักษร โดยพาโทรลคอนโซลไคลแอนท์จะมีโปรแกรมเชื่อมต่อที่ใช้ในการดึงข้อมูลจาก พาโทรลเอเจนท์ออกมาในรูปแบบตาราง (Excel file) ซึ่งจะแสดงข้อมูลแบบกราฟและข้อมูลดิบ ซึ่งการเชื่อมต่อของส่วนประกอบต่างๆจะเป็นในลักษณะ 3 ชั้น (tiers) ดังรูปที่ 3.5

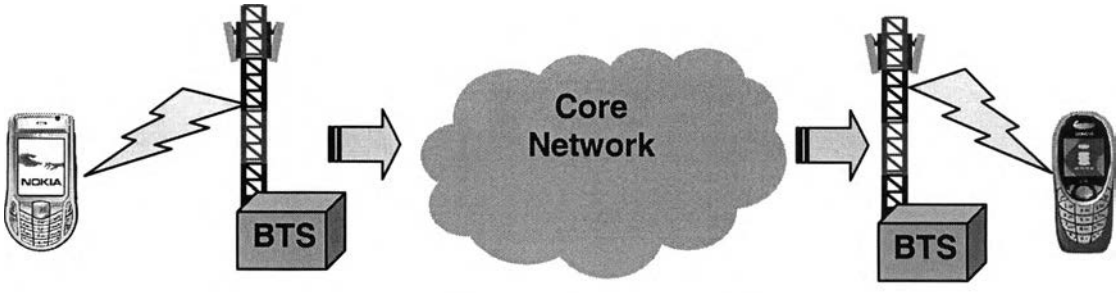


รูปที่ 3.5 แสดงการเชื่อมต่อการทำงานของพาโทรลในลักษณะ 3 ชั้น

จากรูปที่ 3.5 เป็นรูปแสดงการเชื่อมต่อในลักษณะ 3 ชั้น โดยมีการเชื่อมต่อของพาโทรลคอนโซลไคลเอนท์ ที่ต่อผ่านพาโทรลคอนโซลเซิร์ฟเวอร์ เพื่อที่ติดต่อไปยังเซิร์ฟเวอร์ที่มีตัวพาโทรลเอเจนท์ทำงานอยู่

3.1.5 ข้อมูลที่ใช้เป็นกรณีศึกษา

ข้อมูลที่ใช้ทดสอบนี้เป็นข้อมูลของการใช้งานโทรศัพท์เคลื่อนที่ของบริษัทที่ ให้บริการรายใหญ่รายหนึ่งในประเทศไทย กล่าวคือ ในระบบเครือข่ายของการให้บริการโทรศัพท์เคลื่อนที่นี้ จะมีการติดตั้งสัญญาณไว้ตามจุดต่าง ๆ ทั่วทุกภาคของประเทศไทย หรือที่เรียกว่า ชุมสาย หน้าที่ของชุมสายนี้จะมีระบบการส่งสัญญาณของโทรศัพท์เคลื่อนที่ เมื่อลูกค้าใช้บริการโทรศัพท์เคลื่อนที่โดยการกดหมายเลขปลายทางและกดปุ่มโทรออกที่เครื่องโทรศัพท์เคลื่อนที่ ในตัวเครื่องของโทรศัพท์เคลื่อนที่นี้ จะมีตัวส่งสัญญาณไปยังชุมสายที่อยู่ในบริเวณใกล้เคียง และชุมสายจะทำการตรวจสอบเลขหมายปลายทาง และทำการส่งสัญญาณไปยังชุมสายของหมายเลขปลายทางนั้น แต่ถ้าหมายเลขปลายทางนั้นเป็นเครือข่ายของโทรศัพท์เครื่องที่ของผู้ให้บริการรายอื่นก็จะส่งสัญญาณต่อไปยังเน็ตเวอร์อื่นต่อไป และเมื่อมีการตอบรับจากชุมสายปลายทางเรียบร้อยแล้ว ระบบจะทำการเชื่อมต่อสัญญาณระหว่างต้นทางและปลายทาง ทำให้ลูกค้าสามารถใช้บริการได้ และในชุมสายจะเกิดการสร้างรายละเอียดของการใช้บริการในระบบเครือข่าย เรียกว่า ซีดีอาร์ (Call detail record : CDR) ซึ่งเป็นรายละเอียดของการใช้งานโทรศัพท์เคลื่อนที่ของลูกค้าทุกคนที่เกิดขึ้นในแต่ละชุมสายทั่วทุกภาคของประเทศไทย การใช้งานของโทรศัพท์เคลื่อนที่ ที่กล่าวมาแล้วนั้นได้แสดงดังรูปที่ 3.6



รูปที่ 3.6 แสดงการใช้งานของโทรศัพท์เคลื่อนที่

รายละเอียดของการใช้งานของโทรศัพท์เคลื่อนที่ ที่นำมาใช้ในการทดสอบนี้เป็นการใช้บริการของระบบพรีเพด คือเป็นสมัครโดยการซื้อบัตรเติมเงินก่อนจึงจะสามารถใช้บริการในเครือข่ายได้ และเมื่อมีการใช้งานระบบจะตัดเงินออกจากมูลค่าของบัตรเติมเงินเท่ากับจำนวนเงินที่ใช้บริการจริง ทันทีที่มีการใช้งาน และหากมีการใช้งานจากมูลค่าของบัตรเติมเงินจนหมด ต้องมี

การซื้อบัตรเติมเงินใหม่ จึงจะสามารถใช้งานในระบบเครือข่ายได้ รายละเอียดของซีดีอาร์ดังกล่าวแสดงในตารางที่ 3.3

ตารางที่ 3.3 แสดงรายละเอียดของข้อมูลการใช้งานของโทรศัพท์เคลื่อนที่

ชื่อของข้อมูล	ชนิดของข้อมูล	รายละเอียดของชื่อของข้อมูล
NETWORK_TYPE	VARCHAR2(3)	ประเภทของเครือข่ายของระบบมือถือของหมายเลขโทรศัพท์เคลื่อนที่ต้นทาง Prepaid : ต้องมีการเติมเงินในระบบก่อนจึงจะใช้งานเครือข่ายได้ Postpaid : เป็นการใช้งานก่อน แล้วทางผู้ให้บริการจะส่งบิลไปเก็บเงินในภายหลัง
CALL_TYPE	VARCHAR2(2)	ประเภทของการโทร 0 : เป็นการโทรออกในเครือข่ายเดียวกัน 3 : เป็นการโทรออกต่างเครือข่าย 2 : เป็นการรับสายในเครือข่ายเดียวกัน 3 : เป็นการรับสายต่างเครือข่าย
CHARGE_TYPE	VARCHAR2(1)	เป็นค่ากำหนดประเภทของซีดีอาร์ 0 : ไม่คิดเงิน หรือ เป็นการโทรฟรี 1 : มีการคิดเงิน ตามจริง
ROAMFLAG	VARCHAR2(1)	เป็นการระบุว่ามีการใช้เครือข่ายของผู้ใช้บริการรายอื่นหรือไม่ รวมถึงการใช้งานในต่างประเทศ 0 : ไม่ได้ใช้เครือข่ายของผู้บริการรายอื่น 1 : ใช้บริการของเครือข่ายรายอื่น
MOBILE_NO	VARCHAR2(15)	หมายเลขของโทรศัพท์เคลื่อนที่ต้นทาง
CALLED_B_NO	VARCHAR2(24)	หมายเลขของโทรศัพท์เคลื่อนที่ปลายทาง
CALL_START_TIME	DATE	วันเวลาเริ่มต้นใช้บริการ
HOUR	NUMBER(2)	ช่วงชั่วโมงของการใช้บริการ1 - 24
DAY_CODE	DATE	วันที่ในการใช้บริการ
CALL_END_TIME	DATE	วันเวลาที่เลิกใช้บริการ

ชื่อของข้อมูล	ชนิดของข้อมูล	รายละเอียดของชื่อของข้อมูล
TIME_BAND	VARCHAR2(2)	ช่วงเวลาในการใช้บริการ P : ช่วงชั่วโมงเร่งด่วน (Peak) 08:00–20:00 น. O : ช่วงชั่วโมงปกติ (Offpaak) 20:01 – 07:59
CALL_DURATION	NUMBER(5)	ระยะเวลาที่ใช้บริการในหน่ายวินาที
PP_CHARGE_AMOUNT	NUMBER(10,2)	จำนวนเงินที่ใช้บริการ
ZONE_A	NUMBER(4)	พื้นที่ในการให้บริการของหมายเลขต้นทาง
REGION_ORIGINATE	VARCHAR2(2)	ภาคในประเทศไทยที่ให้บริการหมายเลขต้นทาง
ZONE_B	NUMBER(4)	พื้นที่ในการให้บริการของหมายเลขปลายทาง
REGION_DISTINATION	VARCHAR2(2)	ภาคในประเทศไทยที่ให้บริการของหมายเลขปลายทาง
TERMINATE_REASON	VARCHAR2(2)	ประเภทของการหยุดการติดต่อในระบบ เช่น 000 : เป็นการหยุดติดต่อโดยปกติ คือลูกค้าวางสาย 100 : เป็นการหยุดการติดต่อเนื่องจาก เงินลูกค้าไม่มี เงินพอที่จะใช้งานอีกต่อไป
BALANCE_AMOUNT	NUMBER(10,2)	ยอดคงเหลือหลักจากหักค่าบริการแล้ว
SERVICE_TYPE	VARCHAR2(1)	ประเภทของการใช้บริการ 1 : ใช้บริการภายในประเทศ 2 : ใช้บริการในต่างประเทศ 3 : ใช้บริการข้อมูล เช่น รับส่งเมล์
RATE	NUMBER(7,2)	เป็นค่าที่เป็นการบอกเรื่องของส่วนลด
FEATURE_CODE	NUMBER(4)	ประเภทของการใช้บริการในกรณีใช้บริการเสริม
NETWORK_TYPE_B_NO	VARCHAR2(3)	ระบบเครือข่ายของหมายเลขปลายทาง
VAS_FEATURE_CODE	VARCHAR2(4)	ประเภทของบริการเสริม
DISTANCE_BAND	VARCHAR2(2)	ประเภทของการโทร
FEATURE_SUBCODE	VARCHAR2(6)	รายละเอียดของบริการเสริม

ชื่อของข้อมูล	ชนิดของข้อมูล	รายละเอียดของชื่อของข้อมูล
		1 : เป็นการโทรฟรี 2 : เป็นการโทรในพื้นที่ 3 : เป็นการโทรในพื้นที่ใกล้เคียงกัน 4 : เป็นการโทรในพื้นที่ห่างไกล
UNITS	NUMBER(5)	ระยะเวลาในการใช้งานคิดเป็นจำนวนนาที
PROCESS_DATE	DATE	วันที่ในการนำข้อมูลเข้าสู่ระบบ
GROSS_AMOUNT	NUMBER(10,2)	จำนวนเงินที่คิดตามจำนวนที่ใช้จริงไม่รวมค่าส่วนลดจากโปรโมชั่น
FILE_SEQ	VARCHAR(2,5)	ชื่อไฟล์ของซีดีอาร์ที่นำเข้าระบบ

เนื่องจากข้อมูลซีดีอาร์นี้เป็นข้อมูลการใช้งานของลูกค้าในระบบพรีเพด ซึ่งระบบนี้มีผู้ใช้บริการเป็นจำนวนมาก ข้อมูลการใช้งานโดยเฉลี่ยใน 1 วันจะมีซีดีอาร์เกิดขึ้นไม่ต่ำกว่า 50,000,000 ระเบียนต่อวัน หรือมีขนาดประมาณ 170 ไบต์ต่อ 1 ระเบียน โดยเฉลี่ยแล้วประมาณ 8.5 กิกกะไบต์ต่อวัน รายละเอียดของข้อมูลที่นำมาทดสอบมีดังนี้

ข้อมูลของวันที่ 15 สิงหาคม 2548 มีจำนวน 52,424,332 ระเบียน

ข้อมูลของวันที่ 16 สิงหาคม 2548 มีจำนวน 59,892,196 ระเบียน

ข้อมูลของวันที่ 17 สิงหาคม 2548 มีจำนวน 54,743,221 ระเบียน

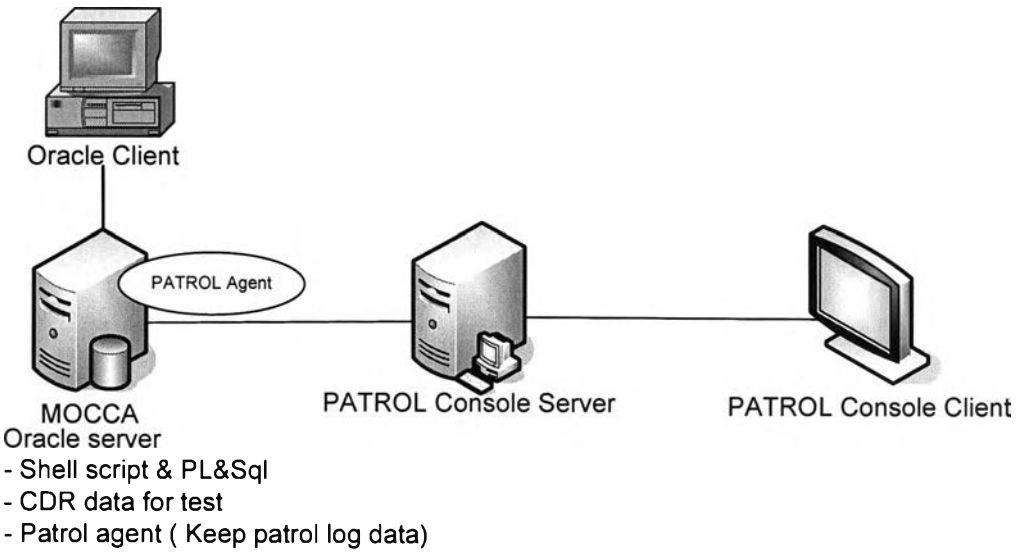
ข้อมูลของวันที่ 18 สิงหาคม 2548 มีจำนวน 55,380,494 ระเบียน

ข้อมูลที่ใช้ในการทดสอบนี้เป็นข้อมูลที่มีปริมาณสม่ำเสมอในทุกวัน ยกเว้น ช่วงวันเสาร์และอาทิตย์ที่มีปริมาณการใช้โทรศัพท์ที่เคลื่อนที่ลดลง ในงานวิจัยนี้จึงเลือกวันที่มีปริมาณการใช้ในช่วงวันทำการปกติเพื่อที่ปริมาณข้อมูลมีผลใกล้เคียงกัน โดยเริ่มจากข้อมูลในวันจันทร์ที่ 15 สิงหาคม 2547 จนถึงวันพฤหัสบดีที่ 18 สิงหาคม 2547 มาเป็นข้อมูลที่ใช้ในการทำวิจัย

จากองค์ประกอบที่ได้กล่าวมาทั้งหมดข้างต้น ในเรื่องของการเตรียมสภาพแวดล้อมที่ใช้ในการวิจัย แสดงดังรูปที่ 3.7 ซึ่งประกอบด้วย ฮาร์ดแวร์ จากรูปก็คือเซิร์ฟเวอร์ที่ชื่อว่า MOCCA เป็นเซิร์ฟเวอร์ที่ใช้ในการติดตั้งระบบฐานข้อมูลออราเคิล และเก็บโปรแกรมที่ใช้ในการทดสอบ ที่เป็นเชลล์สคริปต์ ร่วมกับเอสคิวเอลพลัส รวมทั้งติดตั้งซอฟต์แวร์ที่ใช้ในการวัดสมรรถนะ ซึ่งก็คือพาโทรลเอเจนท์ และสุดท้ายคือเก็บข้อมูลที่นำมาใช้เป็นกรณีศึกษา หรือ ซีดีอาร์ ทุกอย่างจะ

รวมอยู่ในเซิร์ฟเวอร์ MOCCA ในการทำงานนั้นจะใช้การทำงานผ่านไคลเอนส์ จากรูปก็คือ Oracle Client จะเป็นการทำงานบนเวิร์ฟเวอร์โดยผ่านหน้าจอของไคลแอนท์นั้นเอง

และเมื่อทำการทดสอบตามกรณีศึกษาเรียบร้อยแล้ว ข้อมูลของการวัดสมรรถนะของพาโทรลเอเจนท์ที่เกิดขึ้นที่เซิร์ฟเวอร์ MOCCA จะถูกจัดเก็บมาแสดงผลในรูปของข้อมูลตาราง โดยในการจัดเก็บข้อมูลนั้นต้องเข้าหน้าจอของ พาโทรลคอนโซลไคลแอนท์ผ่านไปยังพาโทรลคอนโซลเซิร์ฟเวอร์ เพื่อเข้าไปนำข้อมูลที่เกิดในเซิร์ฟเวอร์ MOCCA ที่อยู่ในรูปของไบนารีไฟล์ มาจัดเก็บใหม่ในรูปแบบของข้อมูลตารางและเก็บไว้ในพาโทรลคอนโซลไคลแอนท์ เพื่อที่การนำข้อมูลของการวัดสมรรถนะนี้มาใช้ในการเปรียบเทียบได้ง่ายขึ้น



รูปที่ 3.7 แสดงภาพสรุประบบที่ใช้ทั้งหมดในการวิจัย

3.2 หลักการวิเคราะห์ข้อมูล

การวิจัยนี้เป็นการนำเสนอ การเปรียบเทียบสมรรถนะการทำงานระหว่าง ระบบจัดการฐานข้อมูลออราเคิล เวอร์ชั่น 9i และเวอร์ชั่น 10g ในด้านของการวัดสมรรถนะนั้นจะพิจารณาในด้านของ เวลาที่ใช้ในการโต้ตอบการทำงาน และการใช้ทรัพยากรต่างๆ เช่น เวลาที่ใช้ในการประมวลผลกลาง ปริมาณการรับส่งข้อมูลที่ใช้ในการอ่านและเขียนข้อมูล หน่วยความจำที่ใช้ในการประมวลผล และ การใช้งานบนจานแม่เหล็กในขณะประมวลผล โดยการศึกษาฟังก์ชั่นการทำงานนั้นได้อ้างอิงจากภาษาของเอสคิวแอล ซึ่งได้แบ่งกลุ่มภาษาได้เป็น 3 กลุ่มคือ

1) **ดีเอ็มแอล** (Data Manipulation Language : DML) เป็นกลุ่มคำสั่งในภาษาเอสคิวแอล ที่ใช้สำหรับการเข้าถึงข้อมูล และการแก้ไขข้อมูล เช่น *Select* ใช้เพื่อคิวรีหาข้อมูล *Insert* ใช้เพื่อเพิ่มข้อมูล *Delete* ใช้เพื่อลบข้อมูล และ *Update* ใช้เพื่ออัพเดตลงในตารางข้อมูล

2) **ดีดีแอล** (Data Definition Language : DDL) เป็นกลุ่มคำสั่งในภาษาเอสคิว-แอลที่ใช้กำหนดออบเจ็กต์ฐานข้อมูล เช่น *Create* ใช้เพื่อเพิ่มตารางลงในฐานข้อมูล *Drop* ใช้เพื่อลบตารางออก และ *Alter* ใช้เพื่อเปลี่ยนแปลงโครงสร้างตารางในฐานข้อมูล

3) **ดีซีแอล** (Data Control Language : DCL) เป็นกลุ่มคำสั่งในภาษาเอสคิวแอลที่ควบคุมความปลอดภัยข้อมูล เช่น *Grant* ใช้เพื่อให้สิทธิในการใช้ฐานข้อมูล และ *Revoke* ใช้เพื่อยกเลิกสิทธิในการใช้ฐานข้อมูล

จากภาษาของเอสคิวแอลเหล่านี้ได้นำมาใช้ในการสร้างกรณีศึกษา เพื่อให้เหมาะในการทำการเปรียบเทียบสมรรถนะที่มีการใช้งานจริงในระบบงานประจำวัน โดยมีรายละเอียดเพิ่มเติมในหัวข้อต่อไป คือหัวข้อ 3.2.1 เหตุการณ์ หรือ กรณีศึกษาที่ใช้ในการเปรียบเทียบสมรรถนะ

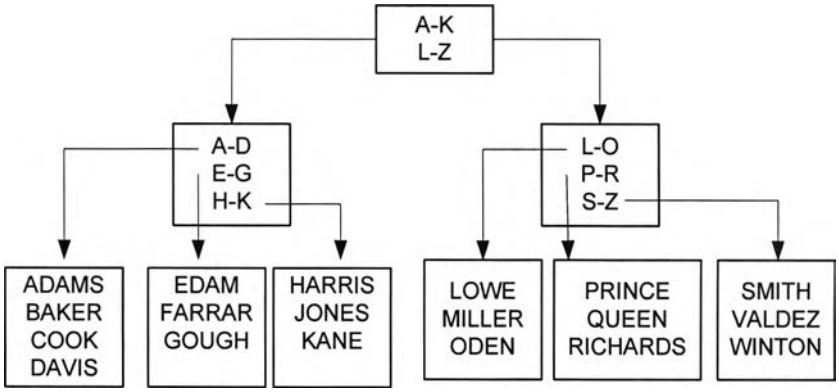
3.2.1 เหตุการณ์ หรือ กรณีศึกษาที่ใช้ในการเปรียบเทียบสมรรถนะ

การเขียนโปรแกรมสำหรับใช้ทดสอบนั้น เป็นการเขียนโปรแกรมเชลล์สคริปต์ ร่วมกับการใช้โปรแกรมเอสคิวพลัส และทำงานบนออราเคิลเซิร์ฟเวอร์ ซึ่งมีรายละเอียดของเหตุการณ์ หรือกรณีศึกษาดังต่อไปนี้

1) การนำข้อมูลเข้าและนำข้อมูลออกจากระบบฐานข้อมูล (Import and Export) การเอ็กซ์พอร์ต คือการนำข้อมูลรวมทั้งโครงสร้างของตารางนั้น มาเขียนเป็นไบนารี่ไฟล์เก็บไว้ ไฟล์ที่ได้นี้สามารถเก็บเป็นไฟล์สำรองเพื่อนำมาใช้ในภายหลัง ไฟล์ที่ได้นี้เรียกว่า ดัมไฟล์ (Dump file) ส่วนการอิมพอร์ตนั้น คือการนำข้อมูลไบนารี่ที่เป็นดัมไฟล์นั้นเข้าสู่ฐานข้อมูลโดยนำมาทั้งโครงสร้างของตาราง

2) การสร้างตารางโดยใช้คำสั่ง *Create table as select* เป็นการนำข้อมูลจากตารางหนึ่ง เข้าสู่ตารางใหม่ โดยใช้โครงสร้างของตารางต้นแบบทุกประการ

3) การสร้างอินเด็กซ์ (Create index) เพื่อทำให้การเข้าถึงข้อมูลมีความรวดเร็ว และมีประสิทธิภาพในการจัดเรียงดีขึ้น ตามโครงสร้างของระบบจัดการฐานข้อมูลออราเคิลนั้นเป็นการทำงานแบบโครงสร้างต้นไม้สมดุล หรือ บีทรี (B*-Tree : Balance Tree) ซึ่งมีสมรรถนะดีมากสำหรับตารางที่มีขนาดใหญ่ โครงสร้างต้นไม้แบบบีทรีนี้แสดงดังรูปที่ 3.8



รูปที่ 3.8 แสดงบล็อคของอินเด็กซ์ บีทรี

4) การปรับปรุงอินเด็กซ์ (Rebuild-index) เมื่อมีการเพิ่มและลบข้อมูลบ่อยๆ เพื่อให้การเข้าถึงข้อมูลมีประสิทธิภาพ ก็ควรที่มีการปรับปรุง อินเด็กซ์สม่ำเสมอ

5) การนำข้อมูลในตารางปกติไปอยู่ในตารางพาทิชัน และพาทิชันอินเด็กซ์ (Partition table and Partition index) ตารางพาทิชันเป็นตารางข้อมูลที่สามารถแบ่งออกเป็นส่วนย่อยๆ เรียกว่าพาทิชัน แต่ละพาทิชันสามารถแยกเก็บไว้ในเทเบิ้ลสเปซต่างกันได้ ตารางข้อมูลแบบพาทิชันนี้ เหมาะกับการจัดเก็บข้อมูลที่มีขนาดใหญ่ เนื่องจากสามารถกระจายตารางไปเก็บไว้ในเทเบิ้ลสเปซที่ต่างกัน ทำให้โอกาสการสูญเสียข้อมูลทั้งหมดในตารางลดลง

6) การปรับปรุงพาร์ติชั่นอินเด็กซ์ (Rebuild – Partition index) เพื่อให้อินเด็กซ์มีประสิทธิภาพของพาทิชันอินเด็กซ์ควรมีการปรับปรุงอินเด็กซ์สม่ำเสมอเช่นกัน

7) การจัดเก็บข้อมูลประจำวัน (Load Transaction) เป็นการนำเอาข้อมูลที่เป็นเท็กซ์ไฟล์ผ่านโปรแกรมเอสคิวแอลโหลดเดอร์ (โดยการใช้คำสั่ง *Sqlldr*) ลงในตาราง ซึ่งทำให้สามารถนำข้อมูลจากฐานข้อมูลอื่นที่ไม่ใช้ออราเคิล เข้ามาในฐานข้อมูลออราเคิลได้ง่ายขึ้น

8) การใช้ฟังก์ชั่นในการสรุปยอด (Summary) เพื่อทำการสรุปข้อมูลจากฐานข้อมูลโดยใช้คำสั่ง *sum()* จะเป็นการนำเอาค่าข้อมูลทั้งคอลัมพ์มาคำนวณหาผลรวมทั้งหมด

9) การจัดเรียงข้อมูล (Sorting) โดยการใช้คำสั่ง *Group by* และ *Order by* คำสั่ง *Group by* เป็นการแบ่งข้อมูลออกเป็นรายการย่อยๆ เรียกว่าเป็นการเรียกดูข้อมูลแบบกลุ่ม เพราะว่าจะรวมกลุ่มข้อมูลจากคำสั่ง *select* แล้วสร้างเป็นเร็คคอร์ดสรุปเพียงเร็คคอร์ดเดียวให้กับแต่ละกลุ่ม ส่วน *Order by* ใช้เพื่อเรียงผลลัพธ์จากการเรียงดูข้อมูล

10) การเรียกดูข้อมูลในปริมาณมากๆ โดยมีการแบ่งเป็นแบบขนาน (Parallel Query) การนับจำนวนข้อมูล และการเข้าถึงข้อมูลโดยการใช้คำสั่ง *Select count(*) from table* และ *Select /*+ parallel(a,2) */ from table* ในการใช้คำสั่ง *parallel* ในเอสคิวแอลสเตทเม็นทนั้น เป็นการแบ่งการทำงานของออราเคิลโปรเซส ซึ่งจำนวนโปรเซสที่ระบบใช้นั้นขึ้นอยู่กับจำนวนซีพียูของเซิร์ฟเวอร์และขึ้นอยู่กับการกำหนดพารามิเตอร์ที่ชื่อว่า PARALLEL_MAX_SERVERS ในพารามิเตอร์ไฟล์ การทดลองนี้เซิร์ฟเวอร์ที่ใช้มีทั้งหมด 2 ซีพียู จึงมีการกำหนด *parallel = 2*

11) การเรียกดูข้อมูล โดยใช้การเชื่อมต่อตาราง (Join Table) คือเรียกดูข้อมูลจากการเชื่อมตารางสองตาราง โดยการจับคู่เร็คคอร์ดที่สัมพันธ์กันจากตารางสองตาราง โดยมีข้อแม้ว่าคอลัมพ์ในตารางที่หนึ่งต้องจับคู่ได้กับคอลัมพ์ในตารางที่สอง และจะต้องมีค่านั้นในคอลัมพ์ทั้งสองตารางเท่ากัน การเชื่อมตารางสามารถทำได้มากกว่าสองตารางขึ้น ไปโดยการใช้เครื่องหมาย (,) คั่นระหว่างตารางไว้ในประโยค *From* และต้องระบุคอลัมพ์ที่ทำการเชื่อมตารางไว้ในประโยค *Where* ในกรณีที่มีการเชื่อมหลายๆ ตาราง จะต้องใช้ลอจิก *and* ร่วมด้วย

12) การเพิ่มข้อมูลใหม่ปริมาณมาก (Insert Parallel) เป็นการเพิ่มข้อมูลใหม่เข้าไปในตาราง โดยข้อมูลใหม่นั้นจะต้องมีโครงสร้างสัมพันธ์กับ โครงสร้างของตารางที่ต้องการเพิ่มนั้นด้วย และการทำ *parallel* นั้นเพื่อเป็นการแบ่งโปรเซสการทำงานให้สามารถทำการเพิ่มข้อมูลได้เร็วขึ้นอีกด้วย

13) การลบข้อมูลปริมาณมาก (Delete Parallel) เป็นการนำข้อมูลออกจากตาราง โดยมีการระบุเงื่อนไขในการนำข้อมูลออก หากไม่ระบุเงื่อนไขจะเป็นการนำข้อมูลออกจากตารางทั้งหมด

14) การวิเคราะห์ตาราง (Analyze Tables) เป็นการเก็บสถิติที่เกี่ยวข้องกับตารางเช่น จำนวนข้อมูลที่มี จำนวนข้อมูลที่ซ้ำกับ คีย์ต่างๆ เพื่อใช้ในการประเมินเวลาในการทำงานตามชุดคำสั่ง

3.2.2 เกณฑ์ หรือ พารามิเตอร์ที่สนับสนุนในการวัดสมรรถนะ

เมื่อได้กรณีศึกษาเรียบร้อยแล้ว ขั้นต่อไปก็คือต้องหาว่าจะใช้อะไรเป็นตัววัดสมรรถนะ เพื่อเป็นการบ่งบอกว่าสมรรถนะของออราเคิลเวอร์ชั่นไหนมีประสิทธิภาพดีกว่ากัน จึงต้องมีการหา เกณฑ์ หรือ พารามิเตอร์ที่ใช้ในการวัดสมรรถนะ ซึ่งการวิจัยนี้ได้แบ่งเป็น 2 ส่วนคือ เวลาที่ใช้ในการโต้ตอบการทำงาน และการใช้ทรัพยากรต่างๆ เช่น การใช้งานของซีพียู การรับส่งข้อมูลในจานแม่เหล็ก และการใช้งานของหน่วยความจำ โดยในการวิจัยนี้ในส่วนของการใช้ทรัพยากรนั้นได้ใช้ข้อมูลที่ได้จากพาโทรลเอเจนท์ ซึ่งมีรายละเอียดของพารามิเตอร์ต่างๆ ดังนี้

1) เวลาที่ใช้ในการโต้ตอบการทำงาน

โดยใช้หลักการตามทฤษฎีของการวัดสมรรถนะการทำงาน คือเป็นระยะเวลาของการทำงานตั้งแต่เริ่มต้นทำงานจนถึงสิ้นสุดการทำงาน

2) การใช้ทรัพยากร

การใช้ทรัพยากรร่วมกันระหว่าง การใช้งานของซีพียู การรับส่งข้อมูลในจานแม่เหล็ก และการใช้งานของหน่วยความจำ โดยข้อมูลเหล่านี้เป็นข้อมูลของพาโทรลเอเจนท์ และมีการเก็บไว้ในพารามิเตอร์ต่างๆ ซึ่งมีรายละเอียดดังต่อไปนี้

(1) CPUIdleTime พารามิเตอร์แสดงถึงจำนวนเปอร์เซนต์ของเวลาที่ ซีพียูไม่ได้ทำการประมวลผล หรืออยู่ในสภาวะนิ่งเฉย

(2) CPURunQSize พารามิเตอร์แสดงถึงจำนวนโปรเซสที่อยู่ในคิว ที่รอการประมวลผล

(3) CPUSystemTime พารามิเตอร์แสดงให้เห็นจำนวนเปอร์เซนต์ของซีพียูประมวลผลกลาง ที่ถูกใช้ไปในส่วนปฏิบัติการของผู้ใช้ และปฏิบัติงานของระบบ รวมไปถึงการใช้ทรัพยากรของซีพียูที่ถูกใช้ไปในการเรียกใช้เคอร์เนลในชุดคำสั่ง

(4) CPUUserTime พารามิเตอร์แสดงถึงจำนวนเปอร์เซนต์ของเวลาที่ใช้ในการประมวลผลในขณะนั้น ที่ถูกใช้ไปในการปฏิบัติตามคำสั่งของผู้ใช้ และปฏิบัติงานของผู้ใช้

(5) CPUCpuUtil พารามิเตอร์แสดงถึงเปอร์เซนต์การทำงานของซีพียูที่ใช้ในการประมวลผล โดยการทำงานของซีพียูที่ใช้ในการประมวลผลนี้ คำนวณจากเวลาที่ซีพียูที่ใช้ในการประมวลผลในส่วนของเวลาของผู้ใช้งานรวมกับเวลาของระบบ

(6) CPUWio พารามิเตอร์แสดงถึงเปอร์เซนต์ของซีพียูที่ใช้ไปในการรอการป้อนข้อมูล และส่งข้อมูลตอบกลับในการทำงานของระบบปฏิบัติการ ค่าที่สูงขึ้นของซีพียูที่ใช้ไปในการรอการป้อนข้อมูล อาจมีสาเหตุที่เป็นไปได้หลายกรณี เช่น หน่วยความจำไม่เพียงพอเนื่องจาก

เวลาที่ระบบใช้ในหน่วยความจำใกล้ถึงขีดจำกัด ทำให้ปริมาณการแลกเปลี่ยน และการโอนถ่ายข้อมูล การใช้ซีพียูเพิ่มขึ้น หรือ การไร้ประสิทธิภาพของชุดคำสั่งที่ป้อนให้ให้ระบบทำงาน ชุดคำสั่งที่ไม่มีประสิทธิภาพมีผลโดยตรงทำให้การทำงานของซีพียูทำงานมากขึ้น

(7) DSKPercentBusy พารามิเตอร์ แสดงเปอร์เซนต์ของเวลาที่ดิสก์ประมวลผลเพื่อทำงานในเรื่องของการโอนถ่ายข้อมูลเมื่อมีการเรียกใช้งานการผู้ใช้งาน ซึ่งเป็นการทำงานที่แสดงปริมาณการทำงานของดิสก์ ค่าพารามิเตอร์ที่สูงขึ้นนั้นอาจแสดงถึงความคล่องตัวในการทำงานของระบบที่มีการใช้งานที่น้องลง

(8) DSKReadWrite พารามิเตอร์ แสดงถึงจำนวนการอ่านและเขียนเพจ ที่มีการถูกอ่านและเขียนข้อมูลลงบนดิสก์ ต่อหน่วยเวลาเป็นวินาที ซึ่งเป็นการแสดงถึงการทำงานของดิสก์ ค่าพารามิเตอร์ที่สูงขึ้นแสดงถึงความคล่องตัวในการทำงานของระบบที่มีการใช้งานที่น้องลง

(9) MEMFreeMem พารามิเตอร์แสดงถึง 1 (กิโลไบต์) เพจของหน่วยความจำที่สามารถทำงานได้ในระบบ ปริมาณของหน่วยความจำที่ใช้งานได้ในระบบเป็นสิ่งที่ฉุกเฉินของระบบปฏิบัติการหากมีปริมาณต่ำในขณะที่มีแอพพลิเคชั่นทำงานอยู่เป็นจำนวนมาก ระบบปฏิบัติการอาจจะเริ่มทำการแลกเปลี่ยนข้อมูลระหว่างหน่วยความจำหลักและ หน่วยความจำสำรอง

(10) PROCPCPUPerc พารามิเตอร์แสดงถึงปริมาณจำนวนเปอร์เซนต์ของซีพียูที่ใช้ประมวลผลตามคำสั่งของโปรเซสซึ่งเปอร์เซนต์นี้คำนวณจากจำนวนซีพียูทั้งหมดในระบบทั้งที่กำลังใช้งานและไม่ได้ใช้งาน

(11) PROCPPMem พารามิเตอร์แสดงถึงปริมาณของหน่วย ความจำเสมือนจริงที่โปรเซสกำลังใช้งานอยู่

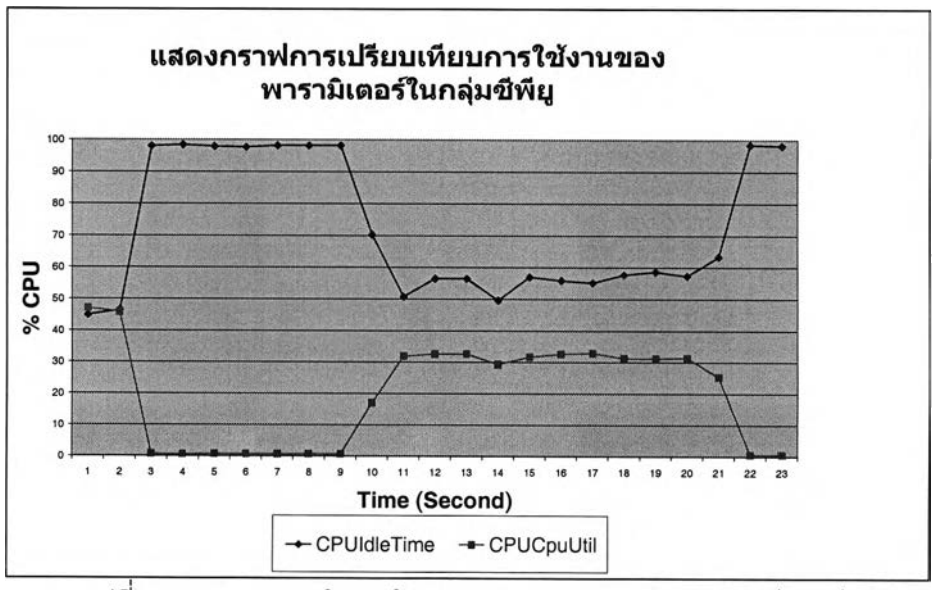
ข้อมูลของพารามิเตอร์เหล่านี้จะเก็บอยู่ในรูปของข้อมูลตาราง โดยข้อมูลจะเก็บไว้ทุกช่วงเวลาในระบบ และสามารถที่นำข้อมูลเหล่านี้มาทำการแสดงในรูปของกราฟ หรือ นำข้อมูลมาแสดงเปรียบเทียบตามช่วงเวลาได้ เช่น ในตารางที่ 3.4 แสดงข้อมูลในรูปแบบของข้อมูลตารางที่มาจากข้อมูลของพาโทรลเอเจนท์

ข้อมูลที่ได้จากพาโทรลเอเจนท์นี้สามารถที่นำข้อมูลที่เป็นพารามิเตอร์ต่างๆ มาทำการเปรียบเทียบกับได้หลากหลาย แต่ในการนำมาเปรียบเทียบนั้นต้องคำนึงถึงค่าความแตกต่างของตัวเลขที่ใช้เป็นตัววัดด้วย หากค่าที่นำมาเปรียบเทียบกันมีความแตกต่างกันมาก กราฟที่แสดงอาจจะไม่ชัดเจน หรือ ไม่ละเอียดเพียงพอ ทำให้ยากต่อการเปรียบเทียบ

ตารางที่ 3.4 แสดงข้อมูลพารามิเตอร์ข้อมูลจากพาโทรลเอเจนท์

Time	CPUIdleTime	Time	CPUCpuUtil
1/30/2006 9:08	44.9083	1/30/2006 9:08	47.0667
1/30/2006 9:09	46.3792	1/30/2006 9:09	45.8917
1/30/2006 9:10	98	1/30/2006 9:10	0.691665
1/30/2006 9:12	98.3917	1/30/2006 9:12	0.591668
1/30/2006 9:13	97.8583	1/30/2006 9:13	0.724998
1/30/2006 9:14	97.6417	1/30/2006 9:14	0.675002
1/30/2006 9:15	98.1917	1/30/2006 9:15	0.641663
1/30/2006 9:16	98.1417	1/30/2006 9:16	0.708333
1/30/2006 9:17	98.125	1/30/2006 9:17	0.670835
1/30/2006 9:19	70.3083	1/30/2006 9:19	17.0417
1/30/2006 9:20	50.675	1/30/2006 9:20	31.8833
1/30/2006 9:21	56.4083	1/30/2006 9:21	32.5167
1/30/2006 9:22	56.3917	1/30/2006 9:22	32.5833
1/30/2006 9:23	49.5667	1/30/2006 9:23	29.2583
1/30/2006 9:24	56.95	1/30/2006 9:24	31.6417
1/30/2006 9:25	55.8667	1/30/2006 9:25	32.5917
1/30/2006 9:26	55.15	1/30/2006 9:26	32.8667
1/30/2006 9:27	57.6333	1/30/2006 9:27	31.25
1/30/2006 9:28	58.6417	1/30/2006 9:28	31.1583
1/30/2006 9:29	57.3	1/30/2006 9:29	31.3833
1/30/2006 9:30	63.35	1/30/2006 9:30	25.2917
1/30/2006 9:31	98.525	1/30/2006 9:31	0.625
1/30/2006 9:32	98.2167	1/30/2006 9:32	0.6

ข้อมูลที่ได้จากทาโทรลเอเจนท์ดังตารางที่ 3.4 สามารถนำมาสร้างให้อยู่ในรูปของกราฟ เพื่อง่ายต่อการเปรียบเทียบ ซึ่งการสร้างกราฟนั้นก็ใช้คุณสมบัติของไฟล์ตาราง แสดงกราฟดังรูปที่ 3.9



รูปที่ 3.9 แสดงการนำเอาข้อมูลตารางมาแสดงเป็นกราฟเปรียบเทียบ

3.3 การเก็บรวบรวมข้อมูล

3.3.1 ขั้นตอนในการเก็บรวบรวมข้อมูลมีการจัดลำดับในการรวบรวมข้อมูลดังนี้

1) เตรียมเซิร์ฟเวอร์และระบบปฏิบัติการของ ซันโซลาริส เวอร์ชั่น 5.8 และติดตั้งพาโทรลเอเจนท์ เวอร์ชั่น 8.6 สำหรับยูนิกซ์

2) ติดตั้งระบบฐานข้อมูลออราเคิล เวอร์ชั่น 9.2.0.1 แบบมาตรฐาน และเริ่มต้นการใช้งานของพาโทรล เอเจนท์

3) สร้างชุดคำสั่งที่ใช้ในการทดสอบ โดยสร้างชุดคำสั่งจากเซลล์สคริปต์ร่วมกับโปรแกรมเอสคิวแอลพลัส จากนั้นทำการทดสอบตามกรณีศึกษา และเก็บรวบรวมข้อมูลจากพาโทรลเอเจนท์ เป็นข้อมูลทดสอบชุดที่ 1

4) เพิ่มประสิทธิภาพการทำงานของระบบฐานข้อมูลออราเคิล เวอร์ชั่น 9.2.0.1 ในส่วนของระบบฐานข้อมูล ในการเพิ่มประสิทธิภาพของระบบฐานข้อมูลออราเคิลเวอร์ชั่น 9.2.0.1 ได้ทำการเพิ่มประสิทธิภาพตามทฤษฎีดังนี้

(1) ปรับเปลี่ยนโครงสร้างของระบบจัดการฐานข้อมูลออราเคิล

ได้ทำการปรับเปลี่ยนเกี่ยวกับหน่วยความจำ โดยการเพิ่มขนาดของหน่วยความจำในระบบฐานข้อมูลออราเคิล หรือ เอสจีเอ ให้เพิ่มขึ้น 25% ซึ่งพารามิเตอร์ที่เกี่ยวข้อง คือ SHARED_POOL_SIZE จากที่ค่ามาตรฐานกำหนดเท่ากับ 117,440,512 ไบท์ (112 เมกกะไบท์) เพิ่มขึ้นเป็น 146,800,640 ไบท์ (140 เมกกะไบท์) ขนาดของหน่วยความจำที่เพิ่มขึ้น 25% นี้นำมาจากการทดลองโดยเพิ่มขนาดของหน่วยความจำครั้งละ 5% และติดตามดูผลของเวลาที่ใช้ในการโต้ตอบ และพบว่าเมื่อเพิ่มขนาดของหน่วยความจำขึ้นจนถึง 25% ผลที่เกิดขึ้นเห็นความแตกต่างที่ชัดเจน และหากเทียบกับจำนวนหน่วยความจำของเซิร์ฟเวอร์ที่มีขนาดหน่วยความจำเท่ากับ 2 กิกกะไบท์ และหน่วยความจำของระบบฐานข้อมูลออราเคิลใช้ไป 140 เมกกะไบท์ เหลือสำหรับระบบปฏิบัติการ 60 เมกกะไบท์ ก็มีขนาดเหมาะสมสำหรับการทำงาน จึงเลือกการเพิ่ม 25% เป็นการวัดสมรรถนะในครั้งนี้

(2) การปรับเปลี่ยนในด้านของโครงสร้างของข้อมูล

- โดยการลดขนาดของของดาต้าไฟล์ จากขนาด 4 กิกกะไบท์ ให้เหลือ 2 กิกกะไบท์ ต่อ 1 ดาต้าไฟล์ เพราะการกำหนดขนาดของดาต้าไฟล์ที่มีขนาดใหญ่จะทำให้การอ่านข้อมูลที่อยู่ในดาต้าไฟล์เดียวกันเป็นแบบเรียงลำดับ หากมีการกำหนดให้ดาต้าไฟล์มีขนาดเล็กลง การเก็บข้อมูลในดาต้าไฟล์จะกระจายอยู่ตามดาต้าไฟล์ต่างๆ ทำให้เป็นการกระจายการอ่านข้อมูล

ไปตามดาต้าไฟล์ที่ต่างกันมีผลให้การอ่านนั้นเร็วขึ้น แต่ถ้าการแบ่งดาต้าไฟล์มีขนาดเล็กเกินไปก็จะทำให้การเก็บข้อมูลกระจายมากเกินไป มีผลต่อการอ่านข้อมูลที่ต้องใช้เวลามากขึ้นเช่นกัน

- สร้างอินเด็กซ์เพื่อความรวดเร็วในการเข้าถึงข้อมูลในกรณีของการเชื่อมตาราง เนื่องจากการเชื่อมตารางสองตารางต้องมีคีย์ของทั้งสองตารางที่ใช้ในการเชื่อมกัน หากตารางใดตารางหนึ่ง หรือทั้งสองตารางมีการสร้างอินเด็กซ์ในคีย์ที่ใช้ในการเชื่อมถึงกันนั้น มีผลทำให้การเชื่อมต่อตารางนั้นเร็วขึ้น

(3) การปรับเปลี่ยนในด้านของชุดคำสั่งในการทำงาน

- มีการใช้ *parallel* ในเอสคิวแอลสเตทเม็นท์ เพื่อเพิ่มความเร็วในการทำโปรเซสโดยเป็นการทำงานแบบคู่ขนาน คือการแบ่งโปรเซสการทำงานให้เป็นส่วนๆ ให้สามารถทำงานพร้อมกัน การใช้งานแบบคู่ขนานสามารถทำได้ ดังนี้

จากเดิมใช้คำสั่ง *select count(*) from table A*

เพิ่มคำสั่งเป็น *select /*+ parallel(a,2) */count(*) from table A*

และการแบ่งโปรเซสการทำงานจะเป็นการระบุให้ซีพียูทำงาน ดังนั้นในการแบ่งโปรเซสจึงไม่ควรเกินจำนวนซีพียูของเซิร์ฟเวอร์ที่ใช้งานอยู่ การแบ่งโปรเซสมากเกินจำนวนของซีพียูจึงไม่มีประโยชน์ และในการวิจัยนี้เซิร์ฟเวอร์ที่ใช้มีจำนวนซีพียูเท่ากับ 2 ซพียู การกำหนด parallel จึงใช้ paralle = 2

- มีการระบุการใช้พาทิชันในกรณีที่ต้องการเข้าถึงตารางข้อมูลที่เป็นพาทิชัน แทนการเข้าแบบตามลำดับปกติ เช่น คำสั่งเอสคิวแอลนี้เป็นการเรียกข้อมูลของข้อมูลในวันที่ 15 สิงหาคม 2005 แสดงดังนี้

```
select  network_type_b_no network_destination,
        count(*) transaction,
        sum(pp_charge_amount) sum_amount
from call_detail
where trunc(call_start_time) = '15-aug-2005'
group by network_type_b_no
order by network_type_b_no
```

เนื่องจากตารางข้อมูลนี้มีการสร้างพาทิชันโดยเก็บข้อมูลของแต่ละวันแยกจากกันเพื่อความสะดวกในการเรียกดูข้อมูล ดังนั้นในส่วนของชุดคำสั่งเอสคิวแอล มีการแก้ไขในส่วนของเงื่อนไขเพื่อเปลี่ยนให้เป็นการระบุพาทิชันของข้อมูลวันที่ 15 สิงหาคม 2005 ดังนี้

```
select  network_type_b_no network_destination,
        count(*) transaction,
        sum(pp_charge_amount) sum_amount
from  call_detail partition(call_detail_20050815)
group by network_type_b_no
order by network_type_b_no
```

เพราะการกำหนดพาทิชันที่ต้องการในชุดคำสั่งจะเป็นการกำหนดให้การค้นหาข้อมูลในพาทิชันนั้นเท่านั้น จึงไม่ต้องทำการค้นหาแบบลำดับจึงใช้เวลาในการค้นหาน้อยลง ได้ผลลัพท์เร็ว

- ในกรณีที่มีการเพิ่มข้อมูล หรือ การนำข้อมูลเข้าตาราง (Import table) จะทำการงดใช้อินเด็กซ์ชั่วคราว (Unusable indexes) เพื่อให้ในขณะที่นำข้อมูลเข้าตาราง ไม่ต้องนำข้อมูลเข้าในอินเด็กซ์ ในการนำข้อมูลเข้าอินเด็กซ์นั้นจะต้องมีการเซ็คคีย์เพื่อเรียงลำดับทำให้การเก็บลงอินเด็กซ์ใช้เวลาเพิ่มขึ้น และเมื่อนำข้อมูลเข้าตารางเรียบร้อยแล้ว จึงทำการปรังปรุงอินเด็กซ์ที่หลัง ซึ่งจะใช้เวลาน้อยกว่ามาก

หมายเหตุ ภาคผนวก ก. โปรแกรมและคำสั่งเชลล์สคริปต์ร่วมกับเอสคิวแอลพลัส
ภาคผนวก ข. โปรแกรมและคำสั่งเชลล์สคริปต์ร่วมกับเอสคิวแอลพลัส แบบเพิ่มประสิทธิภาพ

5) จากนั้นทำการทดสอบตามกรณีศึกษา และเก็บรวบรวมข้อมูลจากพาโทรเอเจนท์เป็นข้อมูลทดสอบชุดที่ 2

6) ยกเลิกระบบฐานข้อมูลออราเคิล เวอร์ชั่น 9.2.0.1 (Uninstall)

7) ติดตั้งระบบฐานข้อมูลออราเคิล เวอร์ชั่น 10.1.0.2 ที่มีการกำหนดพารามิเตอร์ไฟล์ และ การใช้งานของดิกส์ เหมือนระบบฐานข้อมูลออราเคิล เวอร์ชั่น 9.2.0.1 ทุกประการ และเริ่มต้นการใช้งานของพาโทรลเอเจนท์

8) ทดสอบข้อมูลตามกรณีศึกษา และเก็บรวบรวมข้อมูลจาก พาโทรลเอเจนท์เป็นข้อมูลทดสอบชุดที่ 3

9) ยกเลิกระบบฐานข้อมูลออราเคิล เวอร์ชั่น 10.1.0.2 และติดตั้งระบบฐานข้อมูลออราเคิลเวอร์ชั่น 10.2.0.1 และทดสอบข้อมูลตามกรณีศึกษา และเก็บข้อมูลจากพาโทรลเป็นข้อมูลทดสอบชุดที่ 4

10) นำข้อมูลทดสอบทั้งหมดมาเปรียบเทียบสมรรถนะ โดยรายละเอียดในการเปรียบเทียบอยู่ในหัวข้อ 3.3.2 ขั้นตอนการเปรียบเทียบข้อมูล

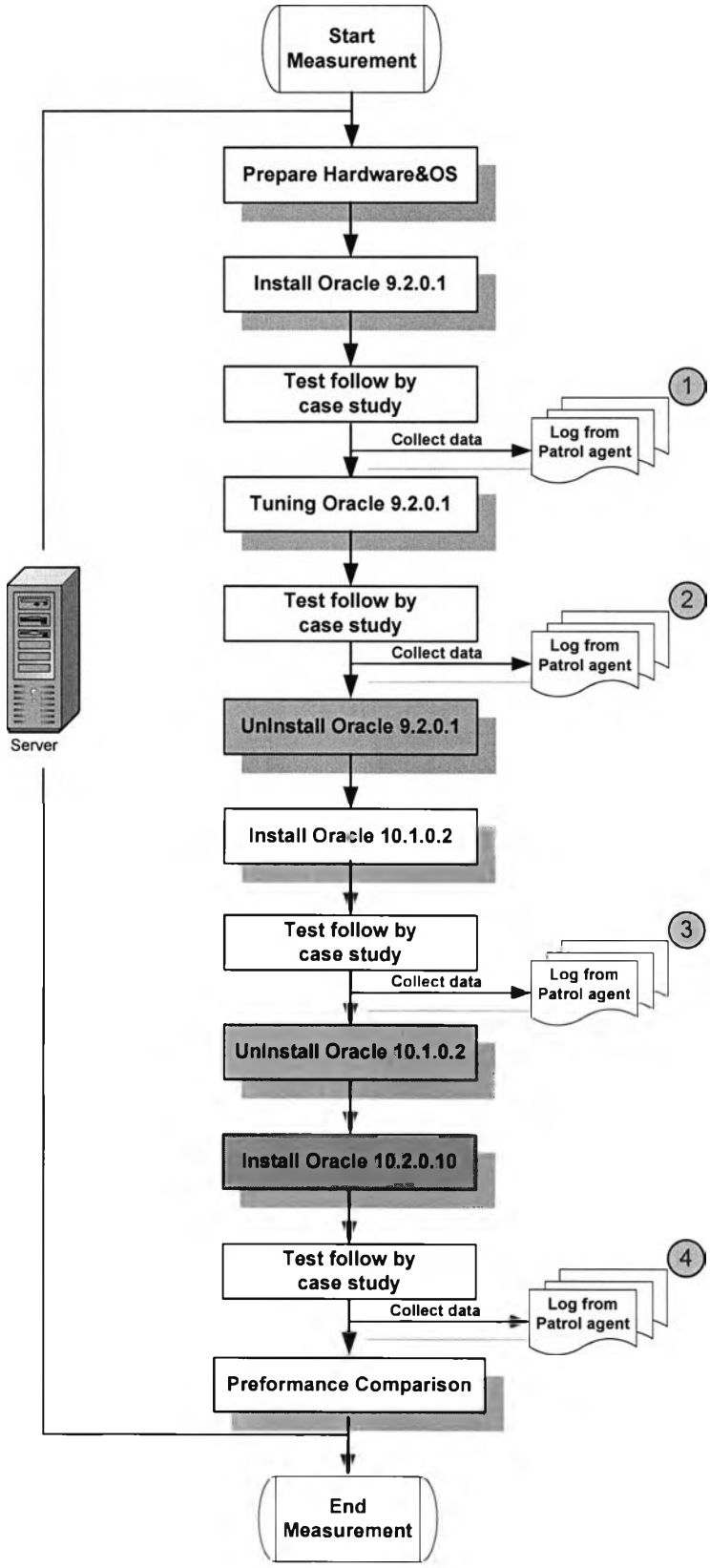
รูปที่ 3.10 แสดงขั้นตอนการเก็บรวบรวมข้อมูลที่กล่าวมาแล้วข้างต้นทั้ง 10 ขั้นตอน

3.3.2 ขั้นตอนการเปรียบเทียบข้อมูล

จากหัวข้อ 3.3.1 ขั้นตอนการเก็บรวบรวมข้อมูล จะเกิดข้อมูลทดสอบทั้งหมด 4 ชุดคือ

1) ข้อมูลชุดทดสอบของระบบฐานข้อมูลออราเคิล เวอร์ชั่น 9.2.0.1
2) ข้อมูลชุดทดสอบของระบบฐานข้อมูลออราเคิลเวอร์ชั่น 9.2.0.1 ที่มีการเพิ่มประสิทธิภาพการทำงาน
3) ข้อมูลชุดทดสอบของระบบฐานข้อมูลออราเคิลเวอร์ชั่น 10.1.0.2
4) ข้อมูลชุดทดสอบของระบบฐานข้อมูลออราเคิล เวอร์ชั่น 10.2.0.1

การนำข้อมูลเปรียบเทียบสมรรถนะนั้นจะนำข้อมูลที่เกิดขึ้นทั้ง 4 ชุด มาทำการเปรียบเทียบพร้อมกัน การวัดสมรรถนะจะพิจารณาในด้านของเวลาที่ใช้ในการโต้ตอบการทำงานและการใช้ทรัพยากรต่างๆ เช่น การใช้งานของซีพียู การรับส่งข้อมูลในจานแม่เหล็ก และการใช้งานของหน่วยความจำ



รูปที่ 3.10 แสดงขั้นตอนของการเก็บรวบรวมข้อมูล