



การประยุกต์สถาปัตยกรรมไคลเอ็นต์เซิร์ฟเวอร์ด้วยระบบจัดการฐานข้อมูล DB2/2

Client/Server Architecture implementation using DB2/2 RDBMS



โดย

นายอัครเดช วัชรภพพงษ์ 35104548

วัน เดือน ปี..... 31 ก.ค. 2540
เลขทะเบียน..... 037087
เลขเรียกหนังสือ..... T 38980 0.1.74 ก

ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร

สาขาวิชาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2538

1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ข้อมูลใดๆ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2538

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การประยุกต์สถาปัตยกรรมคลเ็นต์เซิร์ฟเวอร์ด้วยระบบจัดการฐานข้อมูล DB2/2

ผู้จัดทำ

1 นายอัครเดช วัชรเทพพงษ์ 35104548

..... นพร..... วัชร..... อาจารย์ที่ปรึกษา

(อาจารย์บรรจง ปิยะธำรง)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประยุกต์สถาปัตยกรรมไคลเอ็นต์เซิร์ฟเวอร์ด้วยระบบจัดการฐานข้อมูล DB2/2

นายอัศวเดช วัชรเทพพงษ์ 35104548

อาจารย์บรรจง ปิยะธำรง อาจารย์ที่ปรึกษา

ปีการศึกษา 2538

บทคัดย่อ

ปริญญานิพนธ์นี้นำเสนอแนวทางการประยุกต์สถาปัตยกรรมไคลเอ็นต์เซิร์ฟเวอร์ของระบบจัดการฐานข้อมูล DB2 ของบริษัท IBM โดยมีเซิร์ฟเวอร์เป็นระบบปฏิบัติการ OS/2 Warp Connect และระบบจัดการฐานข้อมูลเชิงสัมพันธ์ DB2/2 เวอร์ชัน 2.1.0 และไคลเอ็นต์เป็นระบบปฏิบัติการบนดอส และวินโดวส์เชื่อมต่อไปยังเซิร์ฟเวอร์ด้วยโปรแกรม DB2 Client Application Enabler (DB2 CAE) สำหรับวินโดวส์ และใช้โปรโตคอล TCP/IP เพื่อเป็นโปรโตคอลสื่อสารระหว่างเซิร์ฟเวอร์และไคลเอ็นต์ โดยใช้มาตรฐาน WinSock สำหรับการเชื่อมต่อ ทั้งนี้ DB2 CAE เป็นตัวกำหนดให้ใช้ และเพื่อความง่ายและรวดเร็วในการพัฒนาแอปพลิเคชันฐานข้อมูลบนเครื่องไคลเอ็นต์ จึงเลือกใช้ ODBC (Open Database Connectivity) เพื่อเป็นตัวกลางในการเชื่อมต่อระหว่าง DB2 CAE และแอปพลิเคชันฐานข้อมูล และวิซวลเบสิกเพื่อเป็นเครื่องมือออกแบบและสร้างแอปพลิเคชันฐานข้อมูลได้อย่างรวดเร็ว

โครงการนี้ยังได้สร้างระบบจำลองเพื่อประกอบการศึกษาและวิธีการประยุกต์สถาปัตยกรรมเซิร์ฟเวอร์ โดยทำเป็นระบบการขายปลีก ณ จุดขาย คือสร้างฐานข้อมูลสำหรับรายการการขายบนเครื่องเซิร์ฟเวอร์ และสร้างแบบกรอกรายการการขายสินค้าบนเครื่องไคลเอ็นต์

Client/Server Architecture implementation using DB2/2 RDBMS

Mr. Akkradach Watcharapupong

Mr. Bunjong Piyatamrong Advisor

1995

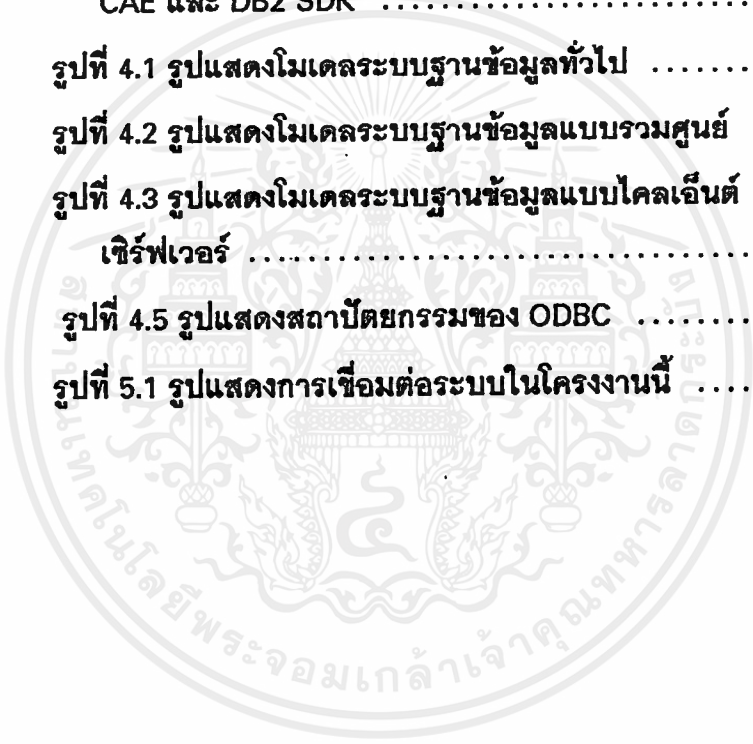
Abstract

This thesis presents the implementation of Client/Server architecture of IBM's DB2 database management system. The server uses OS/2 Warp Connect as its operating system and DB2/2 relational database management system version 2.1.0 and the client uses DOS and Windows. It connects to server by using the DB2 Client Application Enabler for Windows and TCP/IP protocol for being communication protocol between server and client follow the Winsock connect standard , which DB2 CAE use it. And for easy and quick database application development on client workstation, choosing ODBC (Open Database Connectivity) as coordinator between DB2 CAE and database application, and Visual Basic as design and create database application quickly.

This project makes some simulation for study and how to implement the Client/Server Architecture. It is the retail system at point of sale by database for selling list on server and input form for selling on client.

บทที่ 1 บทนำ	หน้า	1
บทที่ 2 สถาปัตยกรรมไคลเอ็นต์เซิร์ฟเวอร์	หน้า	2
2.1 ไคลเอ็นต์เซิร์ฟเวอร์คืออะไร	หน้า	2
2.2 คุณลักษณะของเซิร์ฟเวอร์	หน้า	3
2.3 การให้บริการของระบบปฏิบัติการที่เซิร์ฟเวอร์ต้องการ	หน้า	3
2.4 การให้บริการของระบบปฏิบัติการที่ไคลเอ็นต์ต้องการ	หน้า	4
2.5 รูปแบบสถาปัตยกรรมไคลเอ็นต์เซิร์ฟเวอร์ที่ใช้ในโครงการนี้	หน้า	4
บทที่ 3 DB2/2	หน้า	5
3.1 DB2 คืออะไร?	หน้า	5
3.2 องค์ประกอบหลักของ DB2	หน้า	6
3.3 ส่วนประกอบพื้นฐานของ DB2	หน้า	8
3.4 DB2/2, DB2 CAE และ DB2 SDK สำหรับวินโดวส์	หน้า	9
บทที่ 4 ODBC	หน้า	18
4.1 ODBC คืออะไร	หน้า	18
4.2 สถาปัตยกรรมและโมเดลของ ODBC	หน้า	19
4.3 เปรียบเทียบระหว่างการรวมศูนย์กับไคลเอ็นต์เซิร์ฟเวอร์	หน้า	20
4.4 ลักษณะการโปรแกรมโดยใช้ ODBC	หน้า	21
4.5 สถาปัตยกรรมของ ODBC	หน้า	22
4.6 การเรียกใช้ ODBC อย่างง่ายด้วยวิซวลเบสิก	หน้า	23
4.7 ขนาดของ ODBC	หน้า	23
บทที่ 5 การทดลองเพื่อติดตั้งและเชื่อมต่อระบบ	หน้า	24
5.1 การติดตั้งที่ฝั่งเซิร์ฟเวอร์	หน้า	24
5.2 การติดตั้งฝั่งไคลเอ็นต์	หน้า	28
บทที่ 6 สรุปและวิจารณ์	หน้า	30
ภาคผนวก	หน้า	31
กิตติกรรมประกาศ	หน้า	56
บรรณานุกรม	หน้า	57

รูปที่ 2.1	รูปแสดงสถาปัตยกรรมที่ใช้ในโครงการนี้	หน้าที่ 4
รูปที่ 3.1	รูปแสดงการเข้าถึงเซิร์ฟเวอร์ฐานข้อมูล	หน้าที่ 6
รูปที่ 3.2	รูปแสดงองค์ประกอบของ DB2	หน้าที่ 7
รูปที่ 3.3	รูปแสดงความสัมพันธ์ระหว่าง DB2 กับ DB2 CAE และ DB2 SDK	หน้าที่ 10
รูปที่ 4.1	รูปแสดงโมเดลระบบฐานข้อมูลทั่วไป	หน้าที่ 19
รูปที่ 4.2	รูปแสดงโมเดลระบบฐานข้อมูลแบบรวมศูนย์	หน้าที่ 20
รูปที่ 4.3	รูปแสดงโมเดลระบบฐานข้อมูลแบบไคลเอนต์ เซิร์ฟเวอร์	หน้าที่ 21
รูปที่ 4.5	รูปแสดงสถาปัตยกรรมของ ODBC	หน้าที่ 22
รูปที่ 5.1	รูปแสดงการเชื่อมต่อระบบในโครงการนี้	หน้าที่ 24



บทที่ 1 บทนำ

รายงานฉบับนี้แสดงรายละเอียดขั้นตอนการประยุกต์สถาปัตยกรรมไคลเอ็นต์เซิร์ฟเวอร์ด้วยระบบจัดการฐานข้อมูลเชิงสัมพันธ์ DB2/2 ซึ่งเป็นของบริษัท IBM และรันอยู่บนระบบปฏิบัติการ OS/2 ทั้งนี้เนื่องจากในบริษัทและองค์กรขนาดใหญ่ทั่วไปมักใช้ DB2 นี้เป็นหลัก แต่ความรู้และประสบการณ์ในโปรแกรม DB2 นั้นยังไม่เป็นที่แพร่หลายนัก ผู้เขียนจึงคิดเริ่มศึกษาเพื่อเป็นแนวทางในการทำโครงการหรือแม้กระทั่งทำงานจริงโดยใช้ DB2 เป็นระบบจัดการฐานข้อมูล และเนื่องด้วย DB2/2 เป็นโปรแกรมที่ต้องการระบบที่ค่อนข้างน้อยกว่าตัวอื่นในตระกูลของ DB2 คือรับได้บนแพลตฟอร์มของเครื่องพีซี นอกจากนี้ยังมีความเพียบพร้อมทางด้านภาษา SQL อันถือเป็นผู้นำมาตรฐานรายหนึ่ง แต่ในส่วนของไคลเอ็นต์ต้องทำการสร้างขึ้นเองโดยได้ประยุกต์ใช้ความสามารถของ DB2 CAE บนวินโดวส์ เพื่อต่อเชื่อมไปยังเซิร์ฟเวอร์และใช้โปรโตคอล TCP/IP เป็นตัวสื่อสาร

ในปัจจุบันการพัฒนาแอปพลิเคชันฐานข้อมูลมักใช้มาตรฐาน ODBC ในการเชื่อมต่อระหว่างแอปพลิเคชันเองกับไดร์เวอร์ของฐานข้อมูล ทำให้โครงการนี้ใช้มาตรฐาน ODBC ด้วย เพื่อความอ่อนตัวของแอปพลิเคชันฝั่งไคลเอ็นต์ที่ไม่ต้องยึดติดกับระบบใดระบบหนึ่ง และ DB2 CAE บนวินโดวส์ยังสามารถใช้งานร่วมกับ ODBC ได้อีกด้วย

นอกจากนั้นการพัฒนาแอปพลิเคชันบนวินโดวส์ยังใช้โปรแกรมวิซวลเบสิกในการพัฒนาแอปพลิเคชันฐานข้อมูลเป็นแบบแพลตฟอร์มเพื่อกรอกข้อมูลอีกด้วย ทั้งนี้เพื่อเป็นการแสดงเป็นตัวอย่างในการประยุกต์โดยสมมติเป็นระบบขายปลีก ณ จุดขาย

ผู้เขียนหวังว่าเนื้อหาทั้งหมดนั้นคงเป็นประโยชน์ต่อการใช้งานและความเข้าใจในโปรแกรม DB2/2 ของผู้อ่านไม่มากนักน้อย

นาย อัครเดช วัชรภูพงษ์

ผู้จัดทำ

บทที่ 2 สถาปัตยกรรมไคลเอนต์เซิร์ฟเวอร์

2.1 ไคลเอนต์เซิร์ฟเวอร์คืออะไร

แม้ในปัจจุบัน ไคลเอนต์เซิร์ฟเวอร์ยังคงเป็นคำที่ไม่มีคำจำกัดความแน่ชัด เป็นเพียงความหมายกว้างๆ หากตีความตามชื่อก็คือ ทั้งไคลเอนต์และเซิร์ฟเวอร์ที่แยกออกจากกันทางลอจิกัลแต่ต้องทำงานร่วมกันผ่านระบบเครือข่ายเพื่อให้งานสำเร็จลุล่วง แล้วแตกต่างจากซอฟต์แวร์แบบกระจายศูนย์อย่างไร คงต้องดูคุณสมบัติที่โดดเด่นของไคลเอนต์เซิร์ฟเวอร์ดังนี้

- Service ไคลเอนต์เซิร์ฟเวอร์เน้นความสัมพันธ์ระหว่างโปรเซสที่รันต่างเครื่อง เซิร์ฟเวอร์โปรเซสเป็นผู้รับผิดชอบการให้บริการ ไคลเอนต์โปรเซสก็เป็นผู้ใช้บริการ สิ่งที่สำคัญคือสถาปัตยกรรมไคลเอนต์เซิร์ฟเวอร์จะแยกฟังก์ชันอย่างชัดเจนว่าส่วนใดควรอยู่ฝั่งไคลเอนต์หรือฝั่งเซิร์ฟเวอร์
- Shared resources เซิร์ฟเวอร์สามารถรองรับการให้บริการทรัพยากรแก่ไคลเอนต์หลายตัวได้ในเวลาเดียวกัน
- Asymmetrical protocols ความสัมพันธ์ระหว่างไคลเอนต์และเซิร์ฟเวอร์เป็นแบบ many to one ไคลเอนต์สามารถร้องขอบริการเมื่อใดก็ได้ แต่เซิร์ฟเวอร์ต้องรอให้บริการอยู่เสมอ
- Transparency of location เซิร์ฟเวอร์และไคลเอนต์สามารถอยู่บนเครื่องเดียวกันหรือต่างเครื่องในระบบเครือข่ายก็ได้ ซอฟต์แวร์จะเป็นตัวกันตำแหน่งเซิร์ฟเวอร์ โดยให้บริการผ่านการเรียกฟังก์ชัน
- Mix and Match ในทางอุดมคติ ไคลเอนต์เซิร์ฟเวอร์ซอฟต์แวร์ไม่ขึ้นกับแพลตฟอร์มใดๆ นั่นคือสามารถผสมระบบแบบต่างๆได้
- Message-based exchanges ไคลเอนต์เซิร์ฟเวอร์เป็นระบบที่ผูกพันอย่างหลวม ติดต่อกันโดยกลไกการแลกเปลี่ยนแอสเสจในการร้องขอและตอบรับ
- Encapsulation of services เซิร์ฟเวอร์เป็นผู้เชี่ยวชาญเฉพาะด้าน แอสเสจจะบอกให้เซิร์ฟเวอร์ว่าบริการใดที่ต้องการ เซิร์ฟเวอร์จะจัดการทำงานโดยตัดสินใจว่าต้องทำอะไร นั่นทำให้เปลี่ยนเซิร์ฟเวอร์ได้โดยไม่มีผลกระทบ トラバเท่าที่รูปแบบแอสเสจยังคงเดิม
- Scalability ระบบไคลเอนต์เซิร์ฟเวอร์สามารถขยายได้ทั้งทางแนวนอนและแนวตั้ง โดยที่แนวนอนคือเพิ่มหรือลดไคลเอนต์ได้โดยมีผลกระทบต่อประสิทธิภาพเท่านั้น แนวตั้งคือเพิ่มหรือลดขนาดประสิทธิภาพของเซิร์ฟเวอร์
- Integrity โค้ดและข้อมูลของเซิร์ฟเวอร์ถูกดูแลที่ส่วนกลาง ส่งผลให้ค่าดูแลรักษาถูกลงและปกป้องความถูกต้องของข้อมูลได้ดี ในขณะที่เดียวกันไคลเอนต์ยังคงเป็นส่วนตัวและอิสระ

2.2 คุณลักษณะของเซิร์ฟเวอร์

- รอคอยการร้องขอจากไคลเอนต์
- ปฏิบัติงานความต้องการหลายรายการในเวลาเดียวกัน
- ให้ความสำคัญต่อไคลเอนต์ที่สำคัญมากเป็นพิเศษก่อนเสมอ
- ทำงานแบบฉากหลังได้
- รันอยู่เสมอ
- พร้อมสำหรับการเติบโตใหญ่และฉัวนขึ้นได้ในอนาคต

2.3 การให้บริการของระบบปฏิบัติการที่เซิร์ฟเวอร์ต้องการ

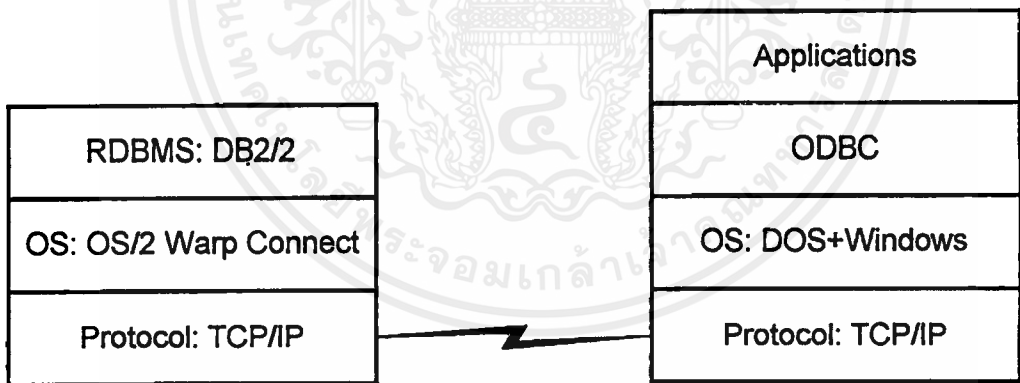
- Task Preemption ระบบปฏิบัติการต้องควบคุมและบังคับการใช้งานซีพียูของโปรเซส
- Task Priority ระบบปฏิบัติต้องให้ความสำคัญแก่บางโปรเซสที่มีค่าไพริออริตี้สูงกว่าก่อน
- Semaphore เพื่อให้โปรเซสที่ใช้ทรัพยากรร่วมกัน และทำงานไปได้โดยไม่เกิดปัญหา
- Interprocess Communication เป็นการรองรับกลไกการแลกเปลี่ยนและใช้ข้อมูลร่วมกันของโปรเซสที่เป็นอิสระต่อกัน
 - Local/Remote Interprocess Communication เพื่อให้กระทำการสื่อสารระหว่างโปรเซสได้โดยไม่ต้องคำนึงถึงว่าอยู่ในระบบเดียวกันหรืออยู่ต่างระบบบนเครือข่าย
 - Thread เป็นหน่วยของการทำงานในโปรเซส เหมาะสำหรับใช้ในโปรแกรมเซิร์ฟเวอร์แบบใช้เหตุการณ์ขับเคลื่อน(event-driven) โดยแต่ละเธรดจะรอคอยเหตุการณ์เฉพาะที่กำหนดไว้ และจะทำงานเมื่อมีเหตุการณ์นั้นเข้ามาเท่านั้น
 - Intertask Protection ระบบปฏิบัติการต้องป้องกันการรบกวนระหว่างโปรเซส และป้องกันไม่ให้ระบบปิดตัวเองลงหรือค้างเนื่องจากโปรเซสใดๆ
 - Efficient Memory Management ระบบจัดการหน่วยความจำต้องมีประสิทธิภาพในการจัดการกับข้อมูลหรือโปรแกรมขนาดใหญ่ได้ โดยการสวอปไปมาระหว่างหน่วยความจำกับดิสก์ใช้บลิทอขนาดเล็ก
 - Dynamically Linked Runtime Extension ส่วนประกอบต่างๆของระบบต้องยืดหยุ่น และสามารถเปลี่ยนเข้าออกได้ โดยไม่ต้องคอมไพล์ระบบใหม่
 - Ubiquitous Communication ระบบปฏิบัติการควรรองรับโปรโตคอลได้หลายชนิดและหลายตัวพร้อมกัน

2.4 การให้บริการของระบบปฏิบัติการที่ไคลเอนต์ต้องการ

- Request/Reply mechanism กลไกการร้องขอและตอบรับไปยังเซิร์ฟเวอร์
- File transfer mechanism. กลไกการถ่ายโอนข้อมูลเพื่อเคลื่อนย้ายรูป หรือข้อความไปมาระหว่างเซิร์ฟเวอร์กับไคลเอนต์
- หากเป็นไปได้ควรทำงานมัลติทาสกิ้งทั้งแบบ preemptive หรือ non-preemptive ก็ได้
- หากเป็นไปได้ควรมีกลไก Interprocess Communication
- ควรมีเซรคคอยรองรับและให้บริการการสื่อสารไปยังเซิร์ฟเวอร์
- ควรมีการป้องกันการรบกวนระบบจากโปรเซส

2.5 รูปแบบสถาปัตยกรรมไคลเอนต์เซิร์ฟเวอร์ที่ใช้ในโครงการนี้

แสดงดังรูปที่ 2.1



บทที่ 3 DB2/2

ผลิตภัณฑ์ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ของ IBM มีอยู่บนระบบปฏิบัติการหลายตัว เช่น MVS, VM, VSE, OS/400, OS/2, AIX, HP-UX, Solaris เป็นต้น ซึ่งผลิตภัณฑ์ทั้งหลายถูกเรียกว่า DB2 หรือ DATABASE 2

3.1 DB2 คืออะไร?

DB2 เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ที่อนุญาตให้ผู้ใช้สร้าง ปรับปรุง และควบคุมฐานข้อมูลเชิงสัมพันธ์ด้วย SQL (Structured Query Language) ได้รับการออกแบบให้เข้ากับความต้องการไม่ว่าปริมาณข้อมูลจะมากหรือน้อยเพียงใด ทั้งยังมีอยู่บนหลายแพลตฟอร์มตั้งแต่ระบบขนาดใหญ่ เช่น MVS/ESA, VM และ VSE ระบบขนาดกลาง เช่น OS/400, AIX และระบบยูนิกซ์ต่างๆ ระบบเครือข่ายท้องถิ่น เช่น OS/2 และวินโดวส์ เป็นต้น ข้อมูลที่ถูกรจัดการโดยเซิร์ฟเวอร์ฐานข้อมูล DB2 สามารถเข้าถึงและจัดการได้โดยแอปพลิเคชันบนเครื่องพีซีที่รันระบบปฏิบัติการ OS/2, ดอสและวินโดวส์ รวมทั้งแอปพลิเคชันที่ถูกพัฒนาบนเครื่องที่รันระบบปฏิบัติการยูนิกซ์ของ IBM, HP และ SUN เป็นต้น การสนับสนุนระบบอื่นยังคงทำต่อไปในอนาคต

DB2 เป็นระบบเปิด (Open System) คือไม่ว่าเป็นไคลเอ็นต์หรือเซิร์ฟเวอร์บนแพลตฟอร์มใดก็สามารถเชื่อมต่อกันได้ โดยใช้โปรโตคอล DRDA (Distributed Relational Database Architecture) ซึ่งทำให้ลดความต้องการอุปกรณ์และเครื่องเคเบิ้ลราคาแพงลงได้ นอกจากนี้ IBM ยังได้เสนอชุดสนับสนุนการเข้าถึงระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS) อื่นที่ใช้โปรโตคอล DRDA นี้แต่ไม่ใช่ DB2 เรียกว่า DDSC (Distributed Database Connection Services)

นอกจากหน้าที่เซิร์ฟเวอร์ฐานข้อมูลแล้ว DB2 ยังได้รวมเครื่องมือเพื่อให้ผู้ใช้สร้างแอปพลิเคชันขึ้นเองสำหรับการเข้าถึงและทำงานกับข้อมูล การสนับสนุนยังรวมถึงการพัฒนาแอปพลิเคชันแบบมัลติมีเดีย และออปเจกตอเรียลไทม์อีกด้วย

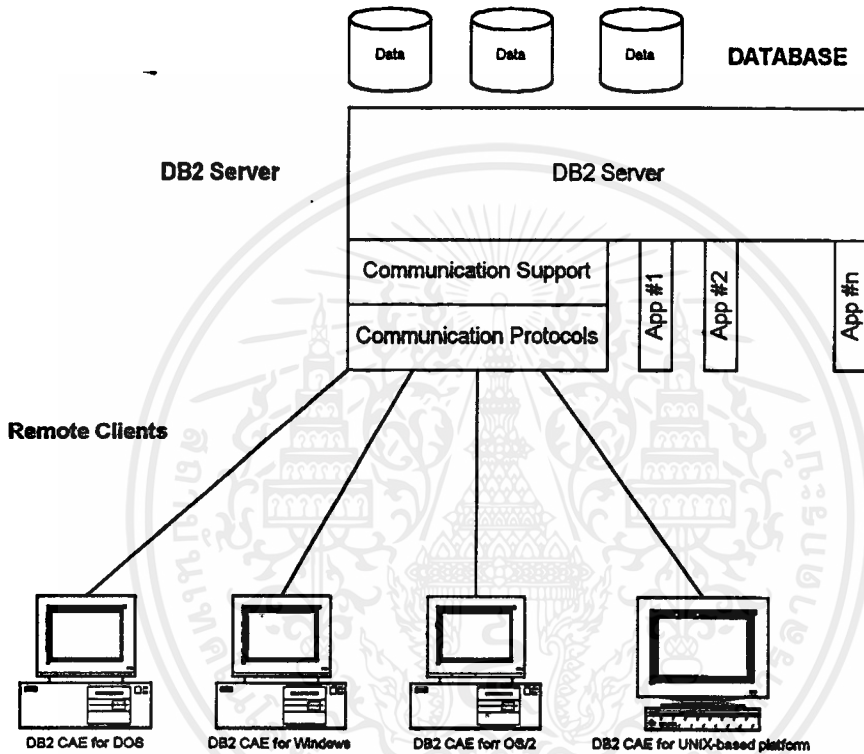
DB2 พื้นฐานมีอยู่ 2 เวอร์ชัน คือ

- DB2 Single-User
- DB2 Server

เอ็นจินฐานข้อมูลของทั้งสองเวอร์ชันเหมือนกัน เอ็นจินคือ ระบบจัดการฐานข้อมูลอันแข็งแกร่งและฟังก์ชันครบครัน รวมถึงการสนับสนุน optimized SQL อันมีพื้นฐานอยู่บนสถิติการใช้งานข้อมูลแท้จริง และเครื่องมือเพื่อช่วยในการจัดการข้อมูล ข้อแตกต่างของทั้งสองเวอร์ชันคือ เครื่องมือบางตัวที่เกี่ยวข้องและความสามารถรองรับรีโมตไคลเอ็นต์

DB2 Single-User สามารถเข้าถึงได้เฉพาะจากแอปพลิเคชันท้องถิ่นที่เครื่องเดียวกันเท่านั้น ซึ่งไม่อยู่ในประเด็นการพิจารณาของโครงการนี้ ที่ต้องการระบบที่สามารถใช้รีโมตไคลเอนต์ได้ด้วย

DB2 Server ได้รวมเอาโค้ดที่ให้ DB2 สามารถรองรับการเข้าถึงจากทั้งไคลเอนต์ท้องถิ่น และรีโมตไคลเอนต์ โดยที่รีโมตไคลเอนต์ต้องติดตั้งชุด DB2 Client Application Enabler (DB2 CAE) อยู่แล้ว เพื่อเข้าถึงเซิร์ฟเวอร์ฐานข้อมูล ดังรูปที่ 3.1



DB2 เซิร์ฟเวอร์เวอร์ชันรวมเอาเอ็นจินฐานข้อมูล, โปรแกรม Database Director, ยูทิลิตี้สำหรับการจัดการระบบและชุดโปรแกรม DB2 CAE เพื่อเข้าถึงรีโมตดาต้าเบสเซิร์ฟเวอร์ หากต้องการสภาพแวดล้อมเพื่อการพัฒนาแอปพลิเคชัน สามารถใช้ DB2 Single-User หรือชุดโปรแกรม DB2 SDK (Software Developer's Kit) ก็ได้ และหากต้องการฐานข้อมูลหลายผู้ใช้ สามารถใช้โปรแกรม Administrator's Toolkit ก็ได้

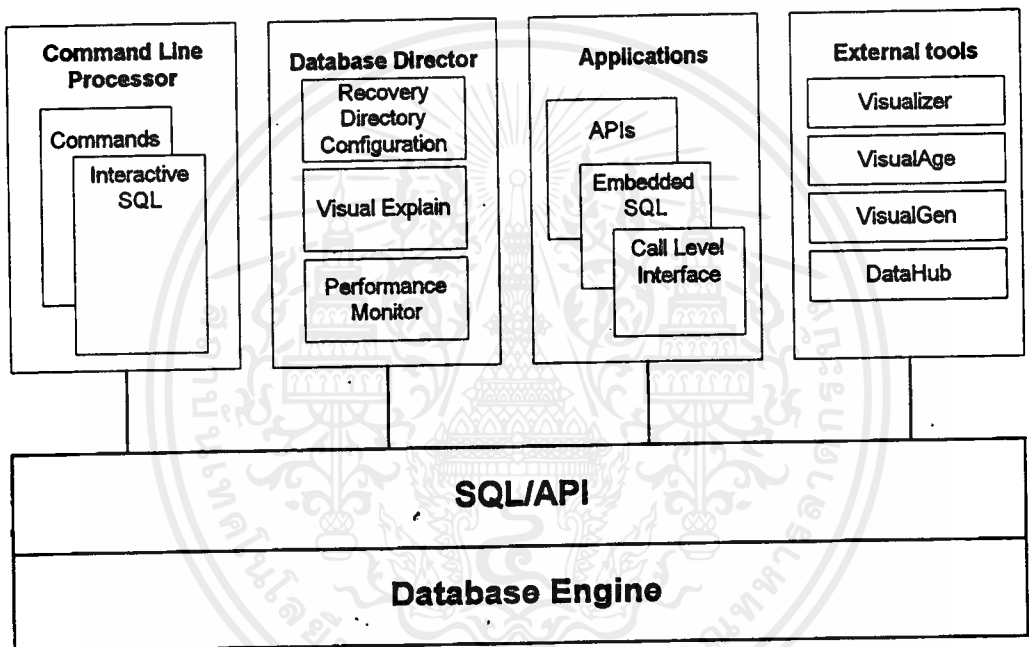
3.2 องค์ประกอบหลักของ DB2

DB2 ให้ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ที่เต็มฟังก์ชันและแข็งแกร่ง พร้อมทั้งกลุ่มของโปรแกรมที่เกี่ยวข้องที่ได้ออกแบบมาสำหรับแพลตฟอร์มของเซิร์ฟเวอร์พื้นฐาน ที่เด่นชัดคือสนับสนุน SQL ของ IBM ใน

โปรแกรมทุกตัวในตระกูลของ DB2 รวมทั้ง DB2 for MVS/ESA, DB2 for OS/400 (DB2/400) และ DB2 for OS/2 (DB2/2)

DB2 รับรองความถูกต้องและข้อกำหนดของฐานข้อมูลทั้งในสภาพแวดล้อมเครื่องเดียวและในระบบเครือข่ายท้องถิ่น ส่วนเพิ่มเติมของ DB2 ก็ยังมีชุดโปรแกรมเพื่อเชื่อมต่อและพีซีคอมพิวเตอร์ที่ทำให้เขียนและติดตั้งแอปพลิเคชันฐานข้อมูลได้

องค์ประกอบหลักของระบบจัดการฐานข้อมูลเชิงสัมพันธ์ DB2 ได้รวมเอ็นจินฐานข้อมูลและโปรแกรมเพื่อเข้าถึงข้อมูลดังเช่น command line processor, Database Director และอื่นๆ ดังรูปที่ 3.2



เอ็นจินฐานข้อมูลให้บริการฟังก์ชันพื้นฐานของระบบจัดการฐานข้อมูลเชิงสัมพันธ์ DB2 จัดการข้อมูล, ควบคุมการเข้าถึงข้อมูล, สร้างแพ็กเกจ, สร้างวิธีการทำงานที่เหมาะสมที่สุด, รองรับการจัดการทรานส์แอ็คชัน รับรองข้อกำหนดของข้อมูล และการปกป้องข้อมูล และรองรับ concurrency control การเข้าถึงข้อมูลทุกทางผ่านทาง SQL สมาชิกพื้นฐานของเอ็นจินฐานข้อมูลคือ ออปเจกฐานข้อมูล, คาทาล็อกของระบบ, ไดรากทอรี และคอนฟิกูเรชันไฟล์

ผู้ใช้สามารถเข้าถึงข้อมูลและจัดการฐานข้อมูลด้วย

- Command line processor รองรับการเข้าถึงข้อมูลผ่านการใช้คำสั่งฐานข้อมูลและ interactive SQL
- Database Director รองรับการติดต่อผู้ใช้แบบกราฟิก เพื่อทำฟังก์ชัน เช่น ูก์คีนฐานข้อมูล, นิยามไดเรกทอรี, ตั้งค่าระบบ, จัดการมีเดีย, Visual Explain และ Performance Monitor

- แอปพลิเคชัน เข้าถึงข้อมูลด้วยการใช้ embedded SQL, Call Level Interface (CLI) หรือ Application Program Interface (API) DB2 รองรับ API หลายตัวเพื่อกระทำการสภาพแวดล้อมฐานข้อมูล
- เครื่องมือภายนอก สามารถรองรับฟังก์ชันที่เพิ่มขึ้นอย่างหลากหลาย โดยใช้ API ของ DB2
- ในเซิร์ฟเวอร์เวอร์ชันของ DB2 มีการสนับสนุนการสื่อสารที่อนุญาตให้เข้าถึงจากเวิร์กสเตชันระยะไกล ในสภาพแวดล้อมเครือข่ายท้องถิ่น

3.3 ส่วนประกอบพื้นฐานของ DB2

ฐานข้อมูลเชิงสัมพันธ์นำเสนอข้อมูลตั้งเป็นที่เก็บตารางหลายตาราง แต่ละตารางประกอบด้วยข้อมูลที่จัดเก็บในรูปคอลัมน์และแถวเชิงลจจิคัล โดยในแต่ละฐานข้อมูลได้รวมกลุ่มตาราง System Catalog ไว้เพื่ออธิบายโครงสร้างเชิงลจจิคัลและพีสิคัลของข้อมูล และคอนฟิกรูเรชันไฟล์ที่บรรจุค่าพารามิเตอร์สำหรับฐานข้อมูล รวมทั้ง recovery log เพื่อจัดเก็บข้อมูลการทำทรานเอ็กชันและกู้คืน

ตารางสามารถบรรจุได้ถึง 255 คอลัมน์ จำนวนแถวไม่จำกัด และข้อมูลขนาดสูงสุดถึง 64 กิกะไบต์ ไม่รวมรวมข้อมูลแบบ large object ข้อมูลในตารางเข้าถึงด้วย SQL ที่เป็นภาษามาตรฐานสำหรับการนิยามและกระทำการข้อมูลในระบบฐานข้อมูลเชิงสัมพันธ์

วิว เป็นอีกทางหนึ่งในการนำเสนอข้อมูลจากตารางหนึ่งหรือมากกว่า สามารถอ้างรวมคอลัมน์ทั้งหมดหรือบางคอลัมน์จากตารางที่กำหนด

อินเด็กซ์เป็นกลุ่มของคีย์ซึ่งชี้ไปยังแถวในตาราง มักทำให้การเข้าถึงแถวมีประสิทธิภาพสูง ด้วยการสร้างหนทางเข้าถึงข้อมูลแถวที่ต้องการโดยตรงผ่านทางตัวชี้ ออปติไมเซอริใช้อินเด็กซ์เพื่อหาหนทางที่ไวที่สุดในการเข้าถึงข้อมูล

ฐานข้อมูลสามารถถูกแบ่งออกส่วนๆเรียกว่า เทเบิลสเปส โดยแต่ละเทเบิลสเปสสามารถกระจายอยู่บนที่จัดเก็บได้มากกว่าหนึ่งที่ เมื่อสร้างตาราง สามารถแยกเก็บอินเด็กซ์หรือ large object ในเทเบิลสเปสต่างหากจากเทเบิลสเปสที่จัดเก็บตารางพื้นฐานก็ได้

แพ็กเกจเป็นออปเจคที่รวมข้อมูลทั้งหมดที่ต้องการเพื่อทำงานประโยค SQL เฉพาะจากแฟ้มเดียว ถูกสร้างระหว่างการ binding โปรแกรมแอปพลิเคชัน และเก็บข้อมูลที่ต้องการสำหรับตาต้าเบสแมนเนเจอร์เพื่อเข้าถึงข้อมูลอย่างมีประสิทธิภาพในโปรแกรมหนึ่งๆ

ตาต้าไทป์เป็นการนำเสนอภายในของค่าข้อมูล ผู้พัฒนาแอปพลิเคชันสามารถสร้างไทป์เฉพาะขึ้นได้สำหรับข้อมูลเฉพาะแบบ แต่การเปรียบเทียบหรือกระทำการใดๆอาจต้องนิยามใหม่ด้วย

ทริกเกอร์ เป็นตัวนิยามกลุ่มของการกระทำที่จะถูกปฏิบัติงานโดยการปรับปรุง, แทรก หรือลบ ข้อมูลบนตารางที่กำหนด มักใช้เพื่อตรวจสอบความถูกต้องของข้อมูลก่อนนำเข้า ให้ค่าใหม่จากแถวที่เพิ่งแทรกเข้ามา เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้น การใช้งานทริกเกอร์ทำให้พัฒนาแอปพลิเคชันได้เร็วขึ้น เนื่องจากไม่ต้องเขียนโค้ดในแต่ละแอปพลิเคชันเพื่อทำในสิ่งที่ทริกเกอร์บนฐานข้อมูลทำอยู่แล้ว สามารถกำหนดครั้งเดียวแล้วใช้สำหรับฐานข้อมูลนั้นได้เลย ไม่ว่าจะ เป็นแอปพลิเคชันตัวใดก็ตาม

ออปติไมเซอร์ SQL ของ DB2 เป็นออปติไมเซอร์แบบ cost-based คือ เลือกหนทางที่มีประสิทธิภาพสูงสุด เพื่อเข้าถึงข้อมูลในตาราง โมดูล Visual Explain ถูกใช้เพื่อวิดู access plan ที่เลือกใช้โดยออปติไมเซอร์

DB2 สร้างและบำรุงรักษากลุ่มของตาราง System Catalog สำหรับแต่ละฐานข้อมูล โดยตารางนั้นบรรจุ ข้อมูลเกี่ยวกับออปเจกฐานข้อมูล เช่น ตาราง, วิว, แพ็กเกจ, ทริกเกอร์ เป็นต้น ซึ่งตาราง System Catalog จะถูก สร้างและปรับปรุงโดยอัตโนมัติ ไม่สามารถสร้างหรือดรอปรได้ แต่สามารถคิวรีและวิวเพื่อความของมันได้

ไดเรกทอรี เป็นส่วนสำคัญในการเข้าถึงฐานข้อมูลทั้งแบบโลคอลและรีโมต ซึ่งมันให้ความมั่นใจว่าการเข้าถึงนั้นโปร่งใส(transparent)สำหรับผู้ใช้และแอปพลิเคชัน ไม่ว่าจะฐานข้อมูลที่แท้จริงอยู่อย่างไรก็ตาม โดยแบ่งเป็น

- ไดเรกทอรีฐานข้อมูล ซึ่งแยกแยะตำแหน่งของฐานข้อมูล ทั้งที่เป็นของระบบ โลคอล และ โหนดอื่น
- โหนดไดเรกทอรี ซึ่งเก็บข้อมูลการเชื่อมต่อเครือข่ายสำหรับรีโมตดาต้าเบส เพื่อต่อไปยังเซิร์ฟเวอร์ฐานข้อมูล

คอนฟิгурेशनไฟล์ เก็บค่าพารามิเตอร์ที่นิยามทรัพยากรที่ถูกจองเอาไว้ให้โปรแกรมชุด DB2 แบ่งเป็น

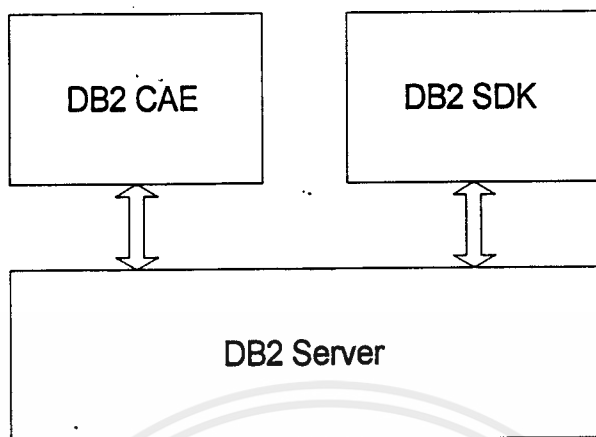
- คอนฟิгурेशनไฟล์ของดาต้าเบสแมนเนเจอร์ เป็นที่เก็บค่าที่ใช้ร่วมกันในชุด DB2
- คอนฟิгурेशनไฟล์ของฐานข้อมูล เป็นที่เก็บค่าในแต่ละโปรแกรมในชุด DB2

3.4 DB2/2, DB2 CAE และ DB2 SDK สำหรับวินโดวส์

ในส่วนนี้อธิบายถึง DB2 และผลิตภัณฑ์ที่เกี่ยวข้องสำหรับแพลตฟอร์มของเครื่องพีซีคือ ดอส, วินโดวส์ และ OS/2 โดยแบ่งออกเป็น

- DB2/2 ซึ่งเป็นเอ็นจินฐานข้อมูลเชิงสัมพันธ์บนระบบปฏิบัติการ OS/2
- DB2 Client Application Enabler (DB2 CAE) สำหรับวินโดวส์ สำหรับการใช้งานที่เครื่องไคลเอ็นต์บนระบบปฏิบัติการดอสและวินโดวส์
- DB2 Software Developer's Kit (DB2 SDK) สำหรับวินโดวส์ สำหรับการพัฒนาแอปพลิเคชันฐานข้อมูลที่เครื่องไคลเอ็นต์บนระบบปฏิบัติการดอสและวินโดวส์

โดยอธิบายความสัมพันธ์ได้ดังรูปที่ 3.3



ในโครงการนี้เครื่องเซิร์ฟเวอร์ใช้ระบบปฏิบัติการคือ OS/2 Warp Connect และระบบจัดการฐานข้อมูล DB2/2 ส่วนเครื่องไคลเอ็นต์นั้นใช้ระบบปฏิบัติการคือ ดอสและวินโดวส์ มีการเชื่อมต่อไปยังเซิร์ฟเวอร์จากเครื่องไคลเอ็นต์ด้วยโปรแกรม DB2 CAE หรือ DB2 SDK สำหรับวินโดวส์

ความสามารถของ DB2 Server

- สนับสนุนระบบจัดการฐานข้อมูลเชิงสัมพันธ์อย่างเต็มรูปแบบ
- สนับสนุนออปติไมเซอร์แบบ cost-based ซึ่งทำให้การคิวรีรวดเร็วยิ่งขึ้น
- รักษาความถูกต้องของข้อมูลในฐานข้อมูล โดยใช้ referential integrity, forward recovery และ multi-level concurrency control เป็นต้น

- สนับสนุนการจัดการฐานข้อมูลขนาดใหญ่อย่างคล่องตัว
- มี command line processor ให้เพื่อใช้งานคำสั่งแบบอินเตอร์แอคทีฟและคำสั่ง SQL
- สนับสนุนแอฟพลิเคชันด้านมัลติมีเดียและออปเจกตเรียนเน็ตโดยสามารถใช้ ทริกเกอร์, User-defined function (UDF), User-defined type (UDT), Constraint, Large Object (LOB) และ recursive SQL เป็นต้น

- สามารถติดตั้งได้ตั้งแต่เครื่องระดับพีซีธรรมดาไปจนถึงระดับซูเปอร์เซิร์ฟเวอร์
- รวมเอาโปรแกรม Database Director เพื่อจัดการระบบของ DB2/2 แบบกราฟิก
- รวมหน้าที่ของ DB2 CAE เข้าไว้ในเซิร์ฟเวอร์แล้วทั้งสำหรับวินโดวส์และ OS/2

- อนุญาตให้ระบบยอมรับความต้องการจากเครื่องไคลเอ็นต์ผ่านระบบเครือข่ายจาก DB2 CAE หรือ DB2 SDK สำหรับบนทุกแพลตฟอร์มของไคลเอ็นต์

- สนับสนุนโปรโตคอลการสื่อสารที่เป็นที่นิยมโดยทั่วไป
- สนับสนุนระบบฐานข้อมูลแบบไคลเอ็นต์เซิร์ฟเวอร์
- สนับสนุนการทำงานของ stored procedure

ความสามารถของ DB2 CAE สำหรับวินโดวส์

DB2

- สนับสนุนการช่วยเหลือการปฏิบัติงานของแอปพลิเคชัน เพื่อเข้าถึงเซิร์ฟเวอร์ฐานข้อมูลของ

- สนับสนุนโปรโตคอลที่เป็นที่นิยมโดยทั่วไป
- สนับสนุน ODBC เพื่อให้แอปพลิเคชันบนวินโดวส์เข้าถึงฐานข้อมูลของ DB2 ผ่าน ODBC
- สนับสนุนการบริหารงานฐานข้อมูลผ่านไปยังเซิร์ฟเวอร์

ความสามารถของ DB2 SDK สำหรับวินโดวส์

- แอปพลิเคชันแบบ embedded SQL
- ออกแบบมาเพื่อสนองความต้องการในการพัฒนาแอปพลิเคชันฐานข้อมูลบนวินโดวส์
 - มี precompiler, library, header file, ตัวอย่างโค้ด และเอกสารต่างๆ สำหรับพัฒนาแอปพลิเคชันแบบ embedded SQL
 - รวมเอา DB2 CAE สำหรับวินโดวส์เข้าไว้ด้วย
 - มี command line processor สำหรับให้ใช้คำสั่งแบบอินเทอร์แอคทีฟและคำสั่ง SQL ได้

DB2 CAE สำหรับวินโดวส์

เป็นตัวสนับสนุนในการปฏิบัติงานแอปพลิเคชันบนวินโดวส์ของเครื่องไคลเอ็นต์ เพื่อเข้าถึงเซิร์ฟเวอร์ฐานข้อมูลเชิงสัมพันธ์ของ DB2 เช่น DB2 for OS/2 (DB2/2), DB2 for AIX รวมถึงต่อผ่านเกตเวย์ไปยังเซิร์ฟเวอร์ฐานข้อมูลเชิงสัมพันธ์ของ DB2 เช่น DB2 for MVS/ESA, DB2 for VSE, DB2 for OS/400 เป็นต้น

การสนับสนุนการปฏิบัติงานนั้น สำหรับแอปพลิเคชันบนวินโดวส์ที่พัฒนาโดยใช้ภาษา C, C++ หรือ COBOL และแอปพลิเคชันที่ใช้ embedded SQL, API ของ Database Manager และ DB2 Call Level Interface (DB2 CLI) นอกจากนั้นยังสนับสนุน ODBC โดยมีไดรเวอร์ให้ เพื่อสนับสนุนแอปพลิเคชันที่พัฒนาขึ้นโดยชุดพัฒนาซอฟต์แวร์ ODBC เวอร์ชัน 1.0 และ 2.0

DB2 CAE ได้รวมยูทิลิตี้ที่ช่วยให้ผู้ใช้สามารถ

- รับแอปพลิเคชันที่ถูกสร้างขึ้นด้วย DB2 SDK สำหรับวินโดวส์หรือเทียบเท่า ซึ่งโปรแกรม อาจประกอบไปด้วย

- Embedded SQL
- Database Manager API
- DB2 CLI API
- เชื่อมแพ็กเกจไปยังฐานข้อมูลเป้าหมาย
- import และ export ข้อมูลจากฐานข้อมูลได้หลายฟอร์แมต
- เก็บข้อมูลของโหนดที่เป็นเซิร์ฟเวอร์และฐานข้อมูลที่เครื่องไคลเอ็นต์สามารถเข้าถึงได้
- เชื่อมต่อไปยังเซิร์ฟเวอร์ระยะไกลผ่านโปรโตคอล IPX/SPX , NETBIOS หรือ TCP/IP ก็ได้
- เข้าถึงระบบความช่วยเหลือสำหรับคำสั่งและแมสเสจต่าง ๆ
- ใช้คำสั่งผ่าน Command line processor ได้ เว้นแต่เป็นการพัฒนาแอปพลิเคชัน หรือคำสั่ง SQL

โปรโตคอลที่สนับสนุน

ความต้องการโปรโตคอลการสื่อสารของเครื่องไคลเอ็นต์ขึ้นอยู่กับเครื่องเซิร์ฟเวอร์ฐานข้อมูลที่ต้องการเชื่อมต่อ สำหรับ DB2 CAE สำหรับวินโดวส์เวอร์ชัน 2 สนับสนุนการเชื่อมต่อไปยังเซิร์ฟเวอร์ของ DB2 for OS/2 เวอร์ชัน 2.1 คือ IPX/SPX, NETBIOS และ TCP/IP

ความต้องการด้านฮาร์ดแวร์

- ซีพียู 386 ขึ้นไป
- หน่วยความจำ 4 เมกาไบต์ขึ้นไป
- เนื้อที่ฮาร์ดดิสก์ 8 เมกาไบต์ขึ้นไปเพื่อติดตั้ง

ความต้องการด้านซอฟต์แวร์

ประกอบด้วยความต้องการคือ ด้านระบบปฏิบัติการและโปรโตคอลสื่อสาร ดังนี้

- ระบบปฏิบัติการดอสและวินโดวส์ 3.1 ขึ้นไป
- โปรโตคอลสื่อสารที่เลือกใช้ในโครงการนี้ คือ TCP/IP โดยอาจใช้
 - IBM TCP/IP สำหรับดอส เวอร์ชัน 2.1.1 หรือสูงกว่า
 - Netware Client สำหรับวินโดวส์และ LAN Workplace Socket driver

ทั้งนี้ DB2 CAE สำหรับวินโดวส์จะทำการเชื่อมต่อผ่าน Socket ไปยังเซิร์ฟเวอร์ของ DB2 โดย WinSock ต้องเป็นเวอร์ชัน 1.1 ขึ้นไป

การสนับสนุน ODBC

เครื่องไคลเอ็นต์สนับสนุนไดรเวอร์ของ ODBC ซึ่งใช้สำหรับแอปพลิเคชันที่พัฒนาด้วย ODBC SDK เวอร์ชัน 1.0 และ 2.0 โดยต้อง

- ติดตั้งไดรเวอร์แมนเนเจอร์ของ ODBC สำหรับวินโดวส์ และ
- ลงทะเบียนไดรเวอร์ ODBC ของ DB2 สูโปรแกรมไดรเวอร์แมนเนเจอร์ ซึ่งไดรเวอร์นี้ ได้รวม

ไว้กับ DB2 CAE แล้ว

การตั้งค่าการสื่อสารสำหรับเครื่องไคลเอ็นต์ด้วยโปรโตคอล TCP/IP

ข้อมูลที่จำเป็นต่อการตั้งค่า

db2node เป็นชื่อของโหนดเซิร์ฟเวอร์ที่ต้องการเชื่อมต่อด้วย โดยตั้งได้ตามความพอใจ แต่ต้องเป็นไปตามกฎการตั้งชื่อและไม่ซ้ำกับของที่มีอยู่แล้วในเครื่องไคลเอ็นต์นั้น เช่น TCHITLER

hostname เป็นชื่อเรียกของเซิร์ฟเวอร์ในระบบเครือข่ายของ TCP/IP ต้องเป็นชื่อไม่สามารถเป็นหมายเลข IP ได้ ตัวเล็กตัวใหญ่มีความแตกต่าง เช่น hitler.ce.kmitl.ac.th

service_name เป็นชื่อของส่วนให้บริการ socket ในโปรโตคอล TCP/IP เช่น db2prj และ db2irq

database_name เป็นชื่อของฐานข้อมูลที่ต้องการเข้าถึง

ขั้นตอนการตั้งค่า

1. ตั้งค่าที่อยู่ในระบบโปรโตคอล TCP/IP ให้เครื่องไคลเอ็นต์เรียกไปยังเซิร์ฟเวอร์ได้ โดยใช้ name server ช่วยหรือเพิ่มหมายเลข IP และชื่อเครื่องลงไปยังแฟ้ม "hosts" ของระบบในเครื่องไคลเอ็นต์ เช่น เพิ่ม

161.246.6.65 hitler.ce.kmitl.ac.th

2. ตั้งค่าตัวแปร environment "DB2COMM" บนเครื่องเซิร์ฟเวอร์ ให้มีค่า "TCPIP" ด้วย แล้วเริ่มโปรแกรมเซิร์ฟเวอร์ของ DB2 ใหม่ เช่นแก้ไข config.sys เพิ่มบรรทัด

set DB2COMM = NPIPE,TCPIP,NETBIOS

3. เพิ่มส่วนให้บริการ Socket ในแฟ้ม services ของทั้งเครื่องไคลเอ็นต์และเครื่องเซิร์ฟเวอร์ เช่น

db2prj 4548/tcp

db2irq

4549/tcp

โดยที่ให้ติดต่อผ่าน Socket หมายเลข 4548 และ 4549 อ้างชื่อเป็น db2prj และ db2irq ตามลำดับ และใช้โปรโตคอล TCP เพื่อสื่อสารกัน

4. ตั้งค่าโปรแกรมเซิร์ฟเวอร์เพื่อให้ใช้บริการ Socket ที่จองไว้โดยออกคำสั่งผ่าน command line processor เช่น

```
update database manager configuration using svcname db2prj
```

ซึ่ง db2prj ได้มาจากข้อ 3 และเพื่อให้การเปลี่ยนแปลงนี้เป็นผลต้องเริ่มเซิร์ฟเวอร์ใหม่อีกครั้ง

5. ตั้งค่าโหนดในเครื่องไคลเอ็นต์โดยใช้คำสั่ง catalog tcpip node ผ่าน command line processor เช่น

```
CATALOG TCP/IP NODE tchitler REMOTE hitler.ce.kmitl.ac.th SERVICE db2prj
```

6. ตั้งค่าฐานข้อมูลประจำโหนดนั้นๆ

หมายเหตุ ในขั้นตอน 5 และ 6 สามารถใช้โปรแกรม DB2 Client Setup ช่วยงานก็ได้

การบริหารผู้ใช้และรหัสผ่านสำหรับเซิร์ฟเวอร์ฐานข้อมูล DB2/2

เซิร์ฟเวอร์ของ DB2 จะอาศัยระบบจัดการผู้ใช้ UPM คือ db2upm.exe ซึ่งจัดการผู้ใช้รหัสผ่านของผู้ใช้ รวมถึงการจัดกลุ่มผู้ใช้อีกด้วย

การ catalog ฐานข้อมูล

ในกรณีที่ต้องการเข้าใช้ฐานข้อมูลในต่างโหนดต้องทำการ catalog ฐานข้อมูลก่อน โดยใช้ Command line processor หรือ โปรแกรม DB2 Client Setup

การใช้งานไคลเอ็นต์

วินโดวส์ไคลเอ็นต์ คือ DB2 CAE และ SDK ทำให้แอปพลิเคชันเข้าถึงฐานข้อมูล ณ โหนดที่ห่างไกล โดยไม่ต้องรู้ตำแหน่งแท้จริงของฐานข้อมูลนั้น ตัวไคลเอ็นต์จะตัดสินใจเรื่องตำแหน่งและจัดการเรื่องความต้องการไปยังเซิร์ฟเวอร์ และรับค่าผลลัพธ์กลับมา

ในการที่จะรับแอปพลิเคชันนั้นก่อนอื่นต้อง

1. catalog ฐานข้อมูลและโหนดก่อน

2. ตรวจสอบการตั้งค่าใน database manager และในแฟ้ม db2.ini ให้ถูกต้อง
3. แน่ใจว่าโปรโตคอลการสื่อสารได้ถูกตั้งค่าพร้อมและรันอยู่แล้ว
4. แน่ใจว่าเซิร์ฟเวอร์ได้ถูกตั้งค่าและรันอยู่แล้ว ซึ่งต้องเริ่มโดยโปรแกรม db2start
5. แน่ใจว่ามีการจัดการความปลอดภัยแล้วบนเซิร์ฟเวอร์ และมีบัญชีผู้ใช้และรหัสผ่านแล้ว
6. เข้าสู่วินโดวส์เพื่อรันโปรแกรม
7. ทำการรันโปรแกรมได้

สำหรับการรันแอปพลิเคชันที่ใช้ ODBC ต้องทำขั้นตอนต่อไปนี้อย่าง

1. แน่ใจว่าได้ติดตั้งไดรเวอร์แมนเนเจอร์ ODBC แล้ว
2. ติดตั้งไดรเวอร์ ODBC ของ DB2 โดยเรียกโปรแกรม db2odbc ซึ่งจะจัดการ
 - เพิ่ม "IBM DB2 ODBC DRIVER" เข้าไปยังส่วนในแฟ้ม odbc.ini
 - คัดลอกแฟ้ม db2odbc.dll และ db2cliw.dll ไปยังไดเรกทอรี system ของวินโดวส์
 - เพิ่มส่วน "IBM DB2 ODBC DRIVER" ในแฟ้ม odbcinst.ini ซึ่งจะทำให้ไดรเวอร์แมนเนเจอร์เข้าใช้ไดรเวอร์ DB2 CLI ได้
3. ลงทะเบียนฐานข้อมูลเข้ากับไดรเวอร์แมนเนเจอร์ ODBC โดยตั้งให้เป็น data source หากแต่ไดรเวอร์แมนเนเจอร์ไม่สามารถอ่านข้อมูลของ DB2 catalog โดยตรงได้ ต้องมีการเรียกโปรแกรมเพื่อคัดลอกแปลงข้อมูลเข้าไปสู่แฟ้ม odbc.ini ดังนี้
 - DB2 Client Setup (เป็นวิธีที่ง่ายสุด)
 - ODBC Administration tool
 - แก้ไขแฟ้ม odbc.ini เอง (ไม่แนะนำ)

สรุปขั้นตอนหากใช้ ODBC คือ

1. catalog ฐานข้อมูลโดยระบุให้เป็น ODBC Data Source ด้วย (มีรูป db2clid.bmp)
2. เรียกโปรแกรม db2odbc เพื่อเชื่อมโยงข้อมูลจาก DB2 Catalog ไปยัง odbc.ini และ odbcinst.ini

การตั้งค่าเครื่องไคลเอ็นต์

แฟ้มที่เก็บค่าของดาต้าเบสแมนเนเจอร์จัดเก็บข้อมูลเกี่ยวกับ DB2 Client Enabler สำหรับเครื่องที่ระบุ โดยมีเพียงหนึ่งแฟ้มในแต่ละการติดตั้งของ client enabler หากแฟ้มที่เก็บค่าหายหรือถูกทำลาย สามารถคืนค่าปริยายโดยคัดลอกแฟ้ม \sqlib\win\cfg\db2sysrq ไปยัง \sqlib\db2\db2system

การดูค่าและตั้งค่าสามารถทำได้โดยใช้

- DB2 Client Setup

- Command line processor ด้วยคำสั่ง

get database manager configuration

reset database manager configuration

update database manager configuration

- API

กฎการตั้งชื่อ

การตั้งชื่อ instance,database,database alias และชื่อ node

- มีตัวอักษรได้ตั้งแต่ 1 ถึง 8 ตัว

- ต้องขึ้นต้นด้วย A-Z,0-9 @,#,\$_(underscore)

การตั้งชื่อผู้ใช้

- มีตัวอักษรได้ตั้งแต่ 1 ถึง 8 ตัว

- สามารถมีอักษรใด ๆ , 0-9 , @ , # , \$

- ไม่สามารถเริ่มต้นด้วย ibm , sys , sql , 0-9 , @ , # , \$

- ไม่สามารถเป็น users , admins , guests , public , local , คำสงวนของ SQL

- ไม่สามารถลงท้ายด้วย \$

การตั้งรหัสผ่าน

- สามารถมีอักขรได้ตั้งแต่ 4 ถึง 8 ตัว

- รวมอักขร A-Z, 0-9, @, #, \$

การตั้งชื่อเครื่องเวิร์คสเตชัน

- สามารถมีอักขรได้ตั้งแต่ 1 ถึง 8 ตัว

- ต้องเริ่มต้นด้วย A-Z, @, #, \$

- ตัวอื่นสามารถเป็น A-Z, 0-9, @, #, \$, &



บทที่ 4 ODBC

เมื่อสืบปีก่อน ความคิดที่จะใช้แอปพลิเคชันเข้าถึงข้อมูล โดยไม่ใช้การโปรแกรมและให้แอปพลิเคชันนั้นเข้าถึงฐานข้อมูลตัวอื่นได้อีก นับเป็นเรื่องที่ยากและแทบเป็นไปได้เลย ODBC ไม่เพียงทำให้เหตุการณ์ดังกล่าวเป็นไปได้ แต่ยังเป็นอย่างง่ายตายอีกด้วย

4.1 ODBC คืออะไร

ODBC หรือ Open Database Connectivity เป็นมาตรฐานของแอปพลิเคชันเพื่อที่จะเข้าถึงข้อมูลของระบบจัดการฐานข้อมูลทั้งแบบเชิงสัมพันธ์และไม่ใช่แบบเชิงสัมพันธ์ ด้วย API ของ ODBC แอปพลิเคชันสามารถเข้าถึงข้อมูลที่เก็บอยู่ในเครื่องหลายระดับตั้งแต่เครื่องพีซี มินิคอมพิวเตอร์และเมนเฟรม แม้ว่าจัดเก็บในรูปแบบที่ต่างกันและมี API ที่ต่างกันไป เพื่อให้บรรลุจุดประสงค์ดังกล่าว สถาปัตยกรรมของ ODBC ต้องแบ่งระดับระหว่างแอปพลิเคชันและ DBMS โดยแอปพลิเคชันรับรู้เพียงการเรียกผ่าน API ของ ODBC เท่านั้น

ODBC มีพื้นฐานอยู่บนข้อกำหนด Call Level Interface (CLI) ของ X/Open SQL Group ซึ่งสถานการณ์โดยทั่วไป คือ

- ซอฟต์แวร์ทุกตัวของไมโครซอฟต์สามารถใช้งาน ODBC ได้ และสนับสนุน ODBC
- การสนับสนุน ODBC ไม่จำกัดเพียงไมโครซอฟต์ แต่ผู้ผลิตรายอื่นก็ให้การสนับสนุนด้วย
- ODBC มีอยู่บนหลายแพลตฟอร์ม เช่น UNIX, OS/2 เป็นต้น

ความต้องการ ODBC เกิดจากปัญหาดังต่อไปนี้

รายใดรายหนึ่ง

- ผู้ใช้ต้องการความอ่อนตัว และเครื่องมือที่ใช้โดยทั่วไปไม่จำกัดเฉพาะฐานข้อมูลของผู้ผลิต

- ผู้ใช้ต้องการแอปพลิเคชันทางธุรกิจที่พัฒนาและแก้ไขได้อย่างรวดเร็ว
- ผู้ใช้ต้องการผลลัพธ์ที่ทำงานได้และเป็นเทคโนโลยีที่เป็นไปได้
- ผู้ใช้ต้องการผลลัพธ์ที่ใช้งานได้ในทุก DBMS
- ผู้ใช้ต้องการประสิทธิภาพสูงสุด
- ผู้ใช้ต้องการความเรียบง่าย

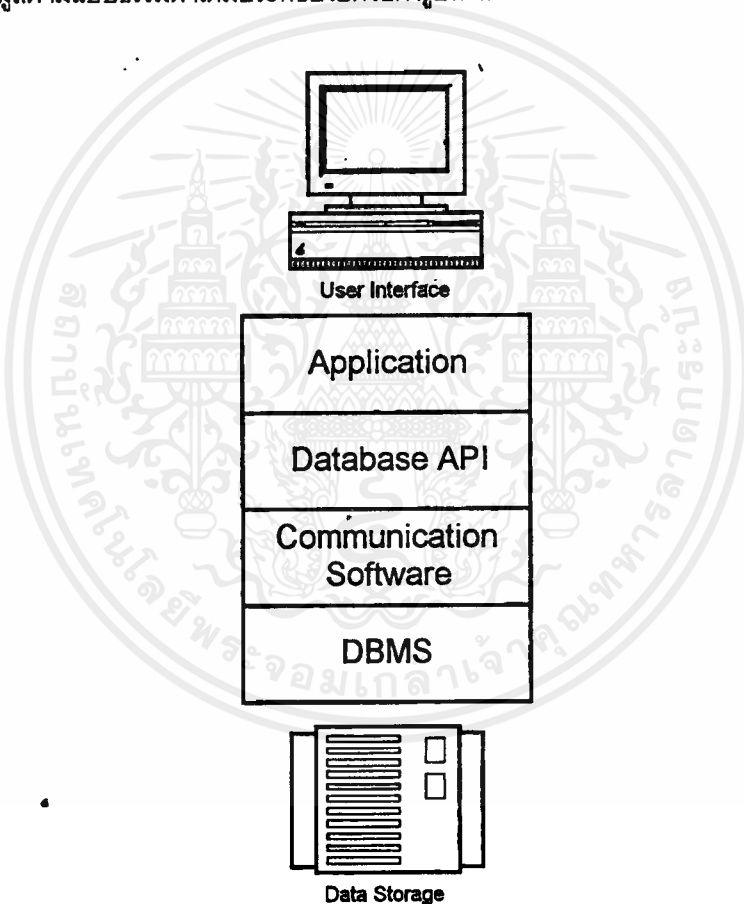
- ผู้พัฒนาไม่ต้องยึดติดกับ DBMS ตัวใดตัวหนึ่ง



- ผู้พัฒนาต้องการแอปพลิเคชันทางธุรกิจโดยใช้เทคนิคการโปรแกรมแบบ RAD (Rapid Application Development) หรือใช้หลักการของวิซวลโปรแกรมมิ่ง
- ผู้พัฒนาต้องการแอปพลิเคชันที่เข้าถึงข้อมูลได้หลายตัว
- ผู้พัฒนาต้องการฟังก์ชันเต็มรูปแบบและประสิทธิภาพสูงสุด

4.2 สถาปัตยกรรมและโมเดลของ ODBC

ฐานข้อมูลตามแบบธรรมดาเดิมประกอบไปด้วยดังรูปที่ 4.1

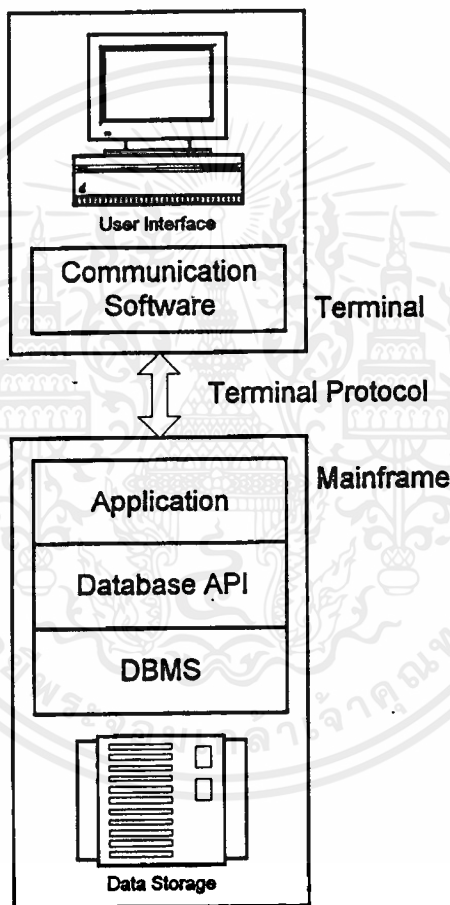


- User Interface เป็นสิ่งที่ผู้ใช้สุดท้าย(End User)เห็นบนหน้าจอ
- Application เป็นโปรแกรมที่ทำให้ผู้ใช้สุดท้ายทำงานได้ลุล่วง โดยใช้ข้อมูลที่ต้องการจากเซิร์ฟเวอร์
- Database API เป็นส่วนโค้ดที่จะขอรับการร้องขอและให้ผลลัพธ์แก่แอปพลิเคชันเมื่อเข้าถึงฐานข้อมูล
- Network/Communication Software เป็นส่วนที่ทำให้โปรแกรมไคลเอ็นต์ติดต่อกับเซิร์ฟเวอร์ได้และใน
ทำงานองกลับกัน

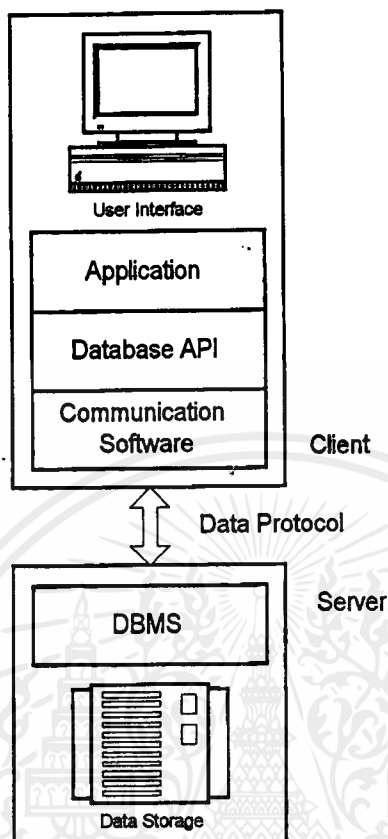
- DBMS เป็นโปรแกรมเซิร์ฟเวอร์เพื่อรองรับความต้องการจากเครื่องไคลเอนต์สำหรับเข้าถึงฐานข้อมูล
- Data Storage เป็นที่ที่ข้อมูลแท้จริงถูกเก็บไว้

4.3 เปรียบเทียบระหว่างการรวมศูนย์กับไคลเอนต์เซิร์ฟเวอร์

ในลักษณะของระบบรวมศูนย์ ดังรูปที่ 4.2



ในลักษณะของระบบไคลเอนต์ เซิร์ฟเวอร์ ดังรูป 4.3



4.4 ลักษณะการโปรแกรมโดยใช้ ODBC

ตามปกติการโปรแกรมในระบบเชิงสัมพันธ์ สามารถทำได้โดยใช้

- Embedded SQL เป็นการเขียนโปรแกรมในภาษาเช่น C , C++ , COBOL โดยผนวกรวมคำสั่งของ SQL ลงไปได้ แต่ต้องผ่านพรีคอมไพเลอร์เสียก่อนจึงทำการคอมไพล์ตามปกติได้ โดยแบ่งเป็น Static SQL และ Dynamic SQL

- Call level Interface เป็นการเขียนโปรแกรมเรียกผ่านฟังก์ชันของ Database API เพื่อทำการดังต่อไปนี้

- เชื่อมต่อไปยังเซิร์ฟเวอร์ฐานข้อมูล
- ส่งประโยค SQL ไปยังเซิร์ฟเวอร์
- ประมวลผล ผลลัพธ์ที่ได้จากการตอบประโยค SQL ที่ส่งให้เซิร์ฟเวอร์ไม่ว่าจะเป็น

ข้อผิดพลาด สถานะข้อมูล หรือข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะทั่วไปของ CLI คล้าย Dynamic SQL แต่ต่างกันว่าไม่มีการใช้พีคอมโพเลอร์ซึ่ง ODBC จะเป็นลักษณะของ Call Level Interface

4.5 สถาปัตยกรรมของ ODBC

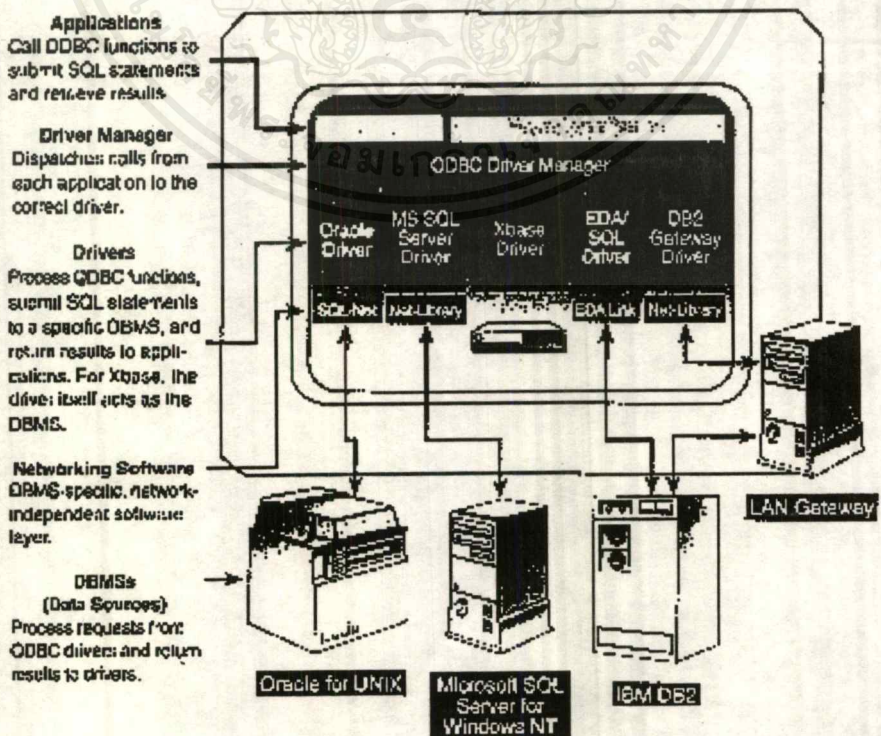
สถาปัตยกรรมของ ODBC ถูกสร้างอยู่บนโมเดลของ ไคลเอ็นต์เซิร์ฟเวอร์ซึ่ง

- สนับสนุน API มาตรฐาน
- ให้ฟังก์ชันครบครันสำหรับในแต่ละ DBMS ที่แตกต่างกัน
- ให้ประสิทธิภาพใกล้เคียงกับ API โดยแท้ของ DBMS

ส่วนประกอบของสถาปัตยกรรม ODBC

- แอปพลิเคชัน
- ไดรเวอร์แมนเนเจอร์ เป็นตัวจัดการไดรเวอร์ทั้งหมดของ ODBC และฐานข้อมูล
- ไดรเวอร์ เป็นตัวกลางในการแปลง ODBC API ให้อยู่ในรูปที่ DBMS เข้าใจ
- ดาต้าซอร์ส เป็นแหล่งข้อมูลที่จะเป็นฐานข้อมูลหลัก

ดังรูปที่ 4.5



4.6 การเรียกใช้ ODBC อย่างง่ายด้วยวิซวลเบสิก

ในโครงการนี้ใช้วิซวลเบสิกในการสร้างแอปพลิเคชันที่ไคลเอ็นต์ ซึ่งต้องมีการผนวกรวมโปรแกรมมางตัวเข้ามาด้วยคือแฟ้มดังต่อไปนี้

- ODBCOR_G.BI เป็นโมดูลประกาศตัวแปรและค่าคงที่แบบ Global ของ ODBC มาตรฐานทั่วไป
- ODBCOR_M.BI เป็นโมดูลประกาศโมดูลของ ODBC มาตรฐานทั่วไป
- ODBEXT_G.BI เป็นโมดูลประกาศตัวแปรและค่าคงที่แบบ Global ของ ODBC ส่วนเสริม
- ODBEXT_M.BI เป็นโมดูลประกาศโมดูลของ ODBC ส่วนเสริม

ซึ่งสามารถทำให้เรียกใช้ ODBC API ได้โดยตรงอย่างสะดวกสบายขึ้น แต่ต้องระมัดระวังเรื่องการผ่านค่าไปยังฟังก์ชัน

4.7 อนาคตของ ODBC

ODBC เวอร์ชัน 2.5 ได้เพิ่มความสามารถหลายด้านเข้าไปดังเช่น

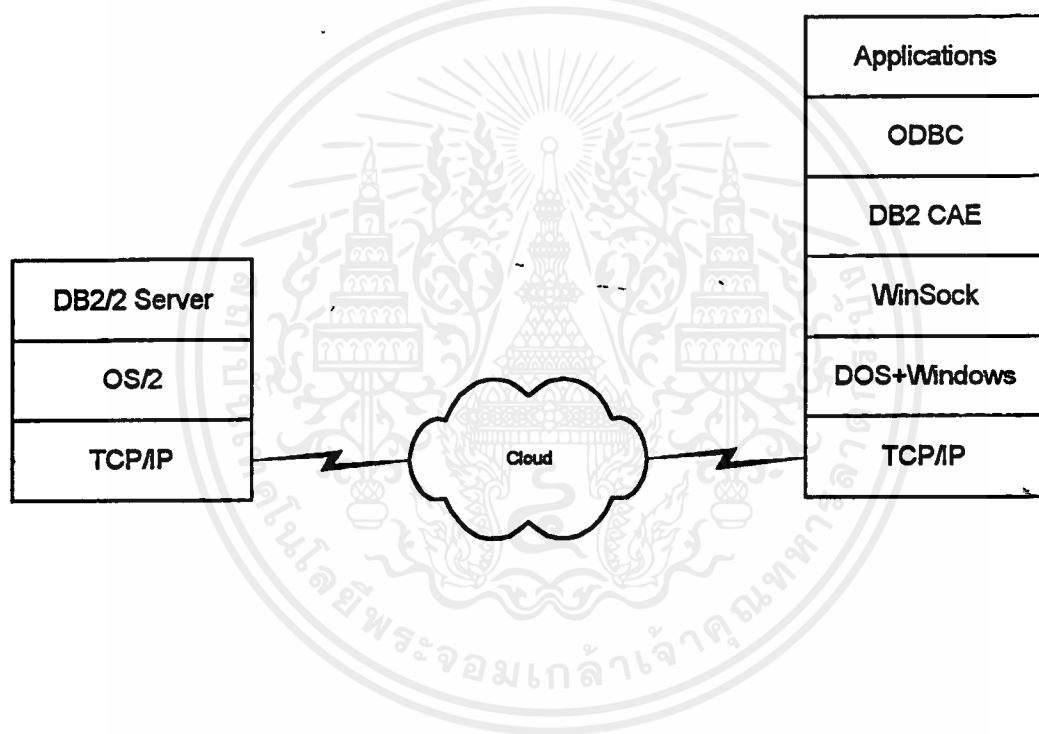
- เพิ่มส่วนการถอนการติดตั้งจากระบบทะเบียนของวินโดวส์ NT และวินโดวส์ 95
- สถาปัตยกรรมที่ดูชัดเจนขึ้น โค้ดต่างๆที่อยู่รวมกันจะอยู่ในแฟ้ม DLL ซึ่งรวมถึงข้อความบ่งบอกความผิดพลาดและ Dialog Box ต่างๆ
- สนับสนุน System Data Source Name ซึ่งให้ผู้ใช้ทุกคนของเครื่องสามารถเข้าถึงการเชื่อมต่อไปยังฐานข้อมูลที่ระบุไว้ได้

ทั้งนี้ ODBC เวอร์ชัน 2.5 มีเฉพาะแบบที่เป็น 32 บิตเท่านั้น ในประมาณปี ค.ศ.1996 ODBC เวอร์ชัน 3.0 ก็จะถูกออกมาโดยเพิ่มความสามารถดังนี้

- จัดให้เข้ากับมาตรฐาน ISO CLI
- สนับสนุนการเข้าถึงข้อมูล OLE
- ปรับเปลี่ยนฟังก์ชันตามความต้องการของผู้พัฒนาแอปพลิเคชัน

บทที่ 5 การทดลองเพื่อติดตั้งและเชื่อมต่อระบบ

ในโครงการนี้ได้ทำการทดลองติดตั้งและเชื่อมต่อระบบต่างๆเข้าด้วยกัน คือ เซิร์ฟเวอร์ที่รันระบบปฏิบัติการ OS/2 Warp Connect และระบบจัดการฐานข้อมูลเชิงสัมพันธ์ DB2/2 เวอร์ชัน 2.1.0 เชื่อมต่อไปยังไคลเอ็นต์ที่รันระบบปฏิบัติการดอสและวินโดวส์ และระบบเชื่อมต่อฐานข้อมูล DB2 CAE สำหรับวินโดวส์ โดยผ่านโปรโตคอลการสื่อสาร TCP/IP ส่วนแอปพลิเคชันในฝั่งไคลเอ็นต์ติดต่อผ่านไปยังเซิร์ฟเวอร์โดยใช้มาตรฐาน ODBC ดังรูปที่ 5.1



5.1 การติดตั้งที่ฝั่งเซิร์ฟเวอร์

ในที่นี้จะกล่าวถึงการติดตั้งและตั้งค่าระบบของ OS/2 Warp Connect และ DB2/2 เวอร์ชัน 2.1.0 ดังมีรายละเอียดและขั้นตอนต่อไปนี้

5.1.1 การติดตั้ง OS/2 Warp Connect

- 1 บุตรจากแผ่น 2 แผ่น ที่โปรแกรมการติดตั้งเตรียมจากซีดีรอม
- 2 เลือกหัวข้อการติดตั้งแบบ Advanced
- 3 เนื่องจากฮาร์ดดิสก์เคยถูกแบ่งพาร์ติชันมาแล้ว โปรแกรมจะทำการติดตั้ง OS/2 ลงบนไดรฟ์

C เลือก Accept the drive

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4 เลือกหัวข้อ Format harddisk แล้วเลือกระบบแฟ้มแบบ FAT

5 รอจนโปรแกรมบอกให้บูตใหม่จากฮาร์ดดิสก์

6 ตั้งค่าอุปกรณ์ต่อพ่วง เช่น เมาส์, การ์ดแสดงผล, เครื่องพิมพ์ เป็นต้น

7 ตั้งค่าระบบและเลือกระบบส่วนที่ต้องการติดตั้งลงไป

7.1 ไม่เลือกกล่อง Multimedia Software Support

7.2 แก้ไขพารามิเตอร์ของ OS/2

- Buffer เป็น 30

- Swappath เป็น CA

8 เข้าสู่การติดตั้งระบบสนับสนุนเครือข่าย

9 เลือกผลิตภัณฑ์ที่จะติดตั้ง คือ

- IBM Peer for OS/2

- IBM TCP/IP for OS/2

10 เลือก

10.1 ออเนปเตอร์ : Eagle Technology NE2000plus

- IO Address เป็น 0300

- IRQ เป็น 10

- Shared RAM Address เป็น 0D000

10.2 OS/2 Peer

- Workstation name เป็น HITLER

- Domain name เป็น COMP_ISO

- Protocol เป็น NetBIOS

10.3 TCP/IP

- Host name เป็น hitler

- IP Address เป็น 161.246.6.65

- Subnet mask เป็น 255.255.255.0

- Router Address เป็น 161.246.6.64

- TCP/IP Domain name เป็น ce.kmitl.ac.th

- Name server เป็น 161.246.6.71

11 โปรแกรมเริ่มทำการตั้งค่าระบบและรีเซตเครื่องเพื่อติดตั้งส่วนสนับสนุนเครือข่าย

12 โปรแกรมบ่งบอกการติดตั้งเสร็จสมบูรณ์แล้ว

13 เมื่อบูตแล้วระบบจะสร้างบัญชีผู้ควบคุมระบบ ใส่ชื่อบัญชี ADEK และรหัสผ่าน

5.1.2 การปรับแต่งค่า OS/2 Warp Connect

1 ปรับแต่ง CONFIG.SYS

- ใส่ rem หน้าบรรทัด IFS=C:\OS2\HPFS.IFS.....

- เพิ่มบรรทัด set RestartObjects=StartupFoldersOnly

- เพิ่มบรรทัด set dircmd=/ogn

- เปลี่ยนค่า FILES เป็น 30

- เปลี่ยนค่า DISKCACHE เป็น 1024,LWAC:C+

- เปลี่ยนค่า Swappath เป็น C:\ 2048 20480

- ใส่ rem หน้าบรรทัด BASEDEV=IBM2FLPY.ADD

- เปลี่ยนค่า DOS เป็น high,umb

2 เพิ่มลบบัญชีผู้ใช้และกลุ่มผู้ใช้ใน User Profile Management (UPM) ดังนี้

- เพิ่มผู้ใช้ ADMIN เป็นผู้ใช้ประเภท Administrator

- ก่อนทำต่อไปให้ logoff จากบัญชี ADEK แล้ว logon เป็นบัญชี ADMIN

- เปลี่ยนบัญชี ADEK เป็นผู้ใช้ประเภท User

- สร้างกลุ่มต่อไปนี้

DBADMIN มีสมาชิกคือ ADEK, ADMIN

DBCTRL, DBMAINT ยังไม่มีสมาชิก

- ลบกลุ่ม GROUPID

3 ปรับแต่งเดสก์ทอป

3.1 การล็อคหน้าจอ

- ให้ Automatic Lockup ภายใน 10 นาที
- ให้ Lock on Startup และใส่รหัสผ่านสำหรับปลดล็อคด้วย

3.2 การ Archive แฟ้มระบบ

- ให้สร้าง archive ทุกครั้งที่บูต

4 ปรับแต่ง TCP/IP

4.1 ส่วน Hostname เพิ่ม Name server address คือ

161.246.4.3

161.246.1.19

4.2 เพิ่มบรรทัดในแฟ้ม CONFIG.SYS คือ

set TZ=GMT+7

4.3 ไม่ควรเลือกให้ start mail ในตอนบูตระบบใหม่

5.1.3 การติดตั้งและตั้งค่า DB2/2

1 เรียกโปรแกรม INSTALLEXE

2 เลือกติดตั้งส่วนประกอบดังนี้

- Server
- Database Director
- Documentation

3 หลังจากติดตั้งโปรแกรมลงฮาร์ดดิสก์เสร็จ ก็ shutdown ระบบ

4 โปรแกรมจะเรียกส่วนการเข้าครั้งแรก โดยควร

- ตั้งค่าผู้ใช้เริ่มต้นที่จะเข้าสู่ DB2 ได้
- สร้างฐานข้อมูล 'SAMPLE' ขึ้นมา

5 ใช้โปรแกรม Database Director ตั้งค่า DB2 โดย

5.1 Administration

- SYSADM เป็น DBADMIN
- SYSCTRL เป็น DBCTRL
- SYSMaint เป็น DBMAINT

5.2 โปรโตคอล

- Workstation name เป็น HITLER
- Service name เป็น db2prj

6 เพิ่มบรรทัดในแฟ้ม services ทั้งใน C:\MPTNETC และ C:\TCPIDOSSETC

```
db2prj 4548/tcp
db2irq 4549/tcp
```

7 เพิ่มบรรทัดในแฟ้ม CONFIG.SYS

```
set db2comm=NPIPE, TCPIP, NETBIOS
```

5.2 การติดตั้งฝั่งไคลเอ็นต์

5.2.1 การติดตั้งโปรแกรม DB2 CAE หรือ SDK สำหรับวินโดวส์

- เรียกโปรแกรม INSTALLEXE
- เลือกส่วนการติดตั้งทั้งหมด
- เลือกไดเรกทอรีเป้าหมาย คือ C:\SQLLIB
- โปรแกรมจะปรับปรุงแฟ้ม AUTOEXEC.BAT ให้อย่างอัตโนมัติ

5.2.2 ติดตั้งไดร์เวอร์เมเนเจอร์ของ ODBC

5.2.3 ติดตั้ง Lan Work Place โดยที่ WinSock ต้องเวอร์ชัน 1.1 หรือสูงกว่า (มีให้ใน WSOCK.ZIP)

5.2.4 ตั้งค่าโปรโตคอล TCP/IP

- เพิ่มบรรทัดในแฟ้ม services

```
db2prj 4548/tcp
db2irq 4549/tcp
```

- เพิ่มบรรทัดในแฟ้ม hosts

161.246.6.65

hitler.ce.kmitl.ac.th

hitler

5.2.5 ตั้งค่าผ่าน command line processor เพื่อให้ใช้โปรโตคอล TCP/IP

```
CATALOG TCP/IP NODE tchitler REMOTE hitler SERVICE db2prj
```

5.2.6 ตั้งค่าฐานข้อมูลประจำโหนด tchitler ด้วยโปรแกรม DB2 Client Setup และตั้งค่าฐานข้อมูลนั้นเป็น Data Source และเรียกโปรแกรม ODBCINST เพื่อให้อ้างอิงด้วย ODBC ได้

หลังจากทำขั้นตอนต่างๆเสร็จสิ้น ก็จะสามารถอ้างอิงถึงฐานข้อมูลผ่านฟังก์ชันของ ODBC ได้โดยใช้ชื่อหรือ Data Source Name (DSN) ตามที่ตั้งไว้ด้วยโปรแกรม Client Setup



บทที่ 6 สรุปและวิจารณ์

6.1 สรุป

โครงการนี้ได้ทำการศึกษาและจัดทำขั้นตอนการติดตั้งโปรแกรมเซิร์ฟเวอร์ฐานข้อมูล DB2/2, ระบบปฏิบัติการพื้นฐานบนเครื่องเซิร์ฟเวอร์ OS/2, ระบบจัดการโปรโตคอลในระบบเครือข่าย TCP/IP, ODBC, DB2 CAE สำหรับวินโดวส์ และการพัฒนาแอปพลิเคชันฐานข้อมูลด้วยวิซวลเบสิก รวมทั้งได้พัฒนาโปรแกรมกรอกแบบฟอร์มการขาย ณ จุดขาย

6.2 วิจารณ์

เนื่องด้วยโปรแกรม DB2/2 ไม่เป็นที่แพร่หลายนัก ข้อมูลต่างๆจึงมีน้อยและผู้เขียนเองก็ไม่มีประสบการณ์ในการทำงานกับ DB2 อีกด้วย จึงทำให้เสียเวลากับการเรียนรู้อย่างมาก ทั้งในส่วนโปรแกรมเซิร์ฟเวอร์ฐานข้อมูลและโปรแกรม DB2 CAE สำหรับวินโดวส์ แต่เมื่อได้ข้อมูลและคำแนะนำจากทางบริษัท IBM ประเทศไทย ก็ได้รับความกระจ่าง

สำหรับการเชื่อมต่อด้วยโปรโตคอล TCP/IP นั้น บนวินโดวส์มีมาตรฐานฟังก์ชันการสื่อสารคือ WinSock ในโครงการนี้จำเป็นต้องใช้งาน WinSock เวอร์ชัน 1.1 แต่ที่ทางภาควิชาฯมีใช้ในระบบเครือข่ายเป็นเวอร์ชัน 1.0 ทำให้ไม่สามารถเชื่อมต่อระหว่างไคลเอ็นต์และเซิร์ฟเวอร์ได้ จนกระทั่งผู้เขียนได้ไปดาวน์โหลด WinSock เวอร์ชัน 1.1 มาจากที่ <ftp://ftp.novell.com/> โดย WinSock ดังกล่าวทั้งหมดอยู่ในโปรแกรมชุด Lan_Work Place ของ Novell

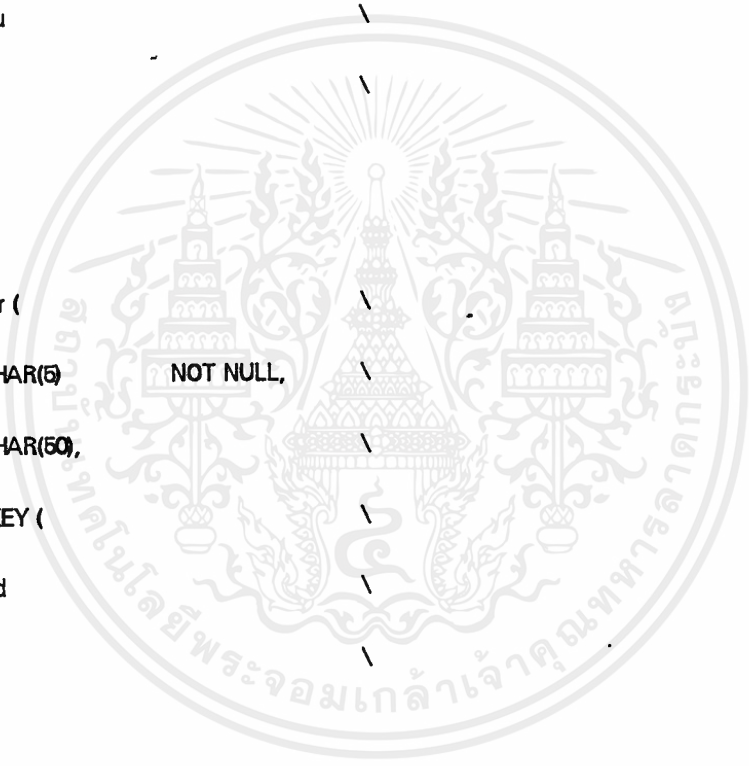


คำสั่ง SQL เพื่อสร้างตารางตัวอย่าง

```
CREATE TABLE goods (
    plu CHAR(15) NOT NULL,
    descrip CHAR(50),
    price DECIMAL(10,2) NOT NULL,
    PRIMARY KEY (
        plu
    )
)
```

```
CREATE TABLE seller (
    sid CHAR(5) NOT NULL,
    descrip CHAR(50),
    PRIMARY KEY (
        sid
    )
)
```

```
CREATE TABLE pos (
    posid CHAR(5) NOT NULL,
    descrip CHAR(50),
    PRIMARY KEY (
        posid
    )
)
```



```

CREATE TABLE selling (
    plu CHAR(10) NOT NULL,
    FOREIGN KEY (plu) REFERENCES goods(plu)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
    sid CHAR(5) NOT NULL,
    FOREIGN KEY (sid) REFERENCES seller(sid)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
    posid CHAR(5) NOT NULL,
    FOREIGN KEY (posid) REFERENCES pos(posid)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
    time TIMESTAMP NOT NULL,
    unit INTEGER NOT NULL,
    PRIMARY KEY (
        plu, sid, posid, time)
)

```

คำสั่ง SQL เพื่อบันทึกข้อมูลตัวอย่าง

```

INSERT INTO goods(plu, descrip, price) \
VALUES ('ABC012345678900', 'OS/2 Warp Connect 3.0 with WinOS2', 7850.00)

INSERT INTO goods(plu, descrip, price) \
VALUES ('ABC012345678901', 'DB2/2 2.1.0 Server', 55700.00)

INSERT INTO goods(plu, descrip, price) \
VALUES ('ABC012345678902', 'DB2 Client Application Enabler for Windows 2.1.0', 1250.00)

INSERT INTO goods(plu, descrip, price) \
VALUES ('ABC012345678903', 'VisualAge 2.0 for OS/2', 11500.00)

INSERT INTO goods(plu, descrip, price) \
VALUES ('ABC012345678904', 'VisualAge 2.0 for Windows', 11000.00)

INSERT INTO seller(sid, descrip) \
VALUES ('S0001', 'Bill Clinton')

INSERT INTO seller(sid, descrip) \
VALUES ('S0002', 'John Mager')

INSERT INTO seller(sid, descrip) \
VALUES ('S0003', 'Boris Yelsin')

INSERT INTO seller(sid, descrip) \
VALUES ('S0004', 'Ko Jong Tong')

INSERT INTO pos(posid, descrip) \
VALUES ('P0001', 'POS of the Alpha shop')

INSERT INTO pos(posid, descrip) \
VALUES ('P0002', 'POS of the Beta shop')

```

```
INSERT INTO pos(posid, descrip) \
VALUES ('P0003', 'POS of the Gamma shop')
```

```
INSERT INTO pos(posid, descrip) \
VALUES ('P0004', 'POS of the Delta shop')
```



ซอร์สโค้ดวิชาเบสิกที่พัฒนาบนฝั่งไคลเอนต์

ชื่อแฟ้ม ADEK.MAK

FRMMAIN.FRM

ODBCOR_G.BI

ODBCOR_M.BI

ODBEXT_G.BI

ODBEXT_M.BI

VBODBC.BAS

MY_G_VAR.BAS

FRMABOUT.FRM

ProjWinSize=74,549,252,203

ProjWinShow=0

IconForm='frmMain'

Title='ADEK's Project'

ExeName='ADEK.EXE'



ชื่อแฟ้ม FRMABOUT.FRM

VERSION 2.00

Begin Form frmAbout

AutoRedraw = -1 'True

BorderStyle = 1 'Fixed Single

Caption = 'About'

ClientHeight = 1920

ClientLeft = 2655

ClientTop = 2415

ClientWidth = 4695

ClipControls = 0 'False

ControlBox = 0 'False

Height = 2325

Left = 2595

LinkTopic = 'Form1'

MaxButton = 0 'False

MinButton = 0 'False

ScaleHeight = 1920

ScaleWidth = 4695

Top = 2070

Width = 4815

Begin PictureBox Picture1

AutoRedraw = -1 'True

Height = 1215

Left = 120

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Picture = FRMABOUT.FRX:0000
 ScaleHeight = 1185
 ScaleWidth = 1185
 TabIndex = 2
 Top = 120
 Width = 1215

End

Begin CommandButton OKButton

Caption = "&OK"
 Height = 495
 Left = 3360
 TabIndex = 0
 Top = 1320
 Width = 1215

End

Begin Label Label1

Caption = 'ADEK's Project for Client/Server architecture implementation using DB2/2'
 FontBold = -1 'True
 FontItalic = 0 'False
 FontName = 'Arial'
 FontSize = 12
 FontStrikethru = 0 'False
 FontUnderline = 0 'False
 Height = 1095
 Left = 1560

```
TabIndex = 1
Top = 120
Width = 3015
```

```
End
```

```
End
```

```
' Active the frmAbout form
```

```
Sub Form_Activate ()
```

```
' Centerized form
```

```
Call CenterWindow(frmMain, frmAbout)
```

```
End Sub
```

```
' Click the OKButton button
```

```
Sub OKButton_Click ()
```

```
' Hide the frmAbout form
```

```
frmAbout.Hide
```

```
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อแฟ้ม FRMMAIN.FRM

VERSION 2.00

Begin Form frmMain

AutoRedraw = -1 'True

BorderStyle = 3 'Fixed Double

Caption = 'Cjient'

ClientHeight = 3960

ClientLeft = 1965

ClientTop = 1530

ClientWidth = 6000

ControlBox = 0 'False

Height = 4365

Left = 1905

LinkTopic = 'Form1'

ScaleHeight = 3960

ScaleWidth = 6000

Top = 1185

Width = 6120

Begin Frame framResult

·Caption = 'Result'

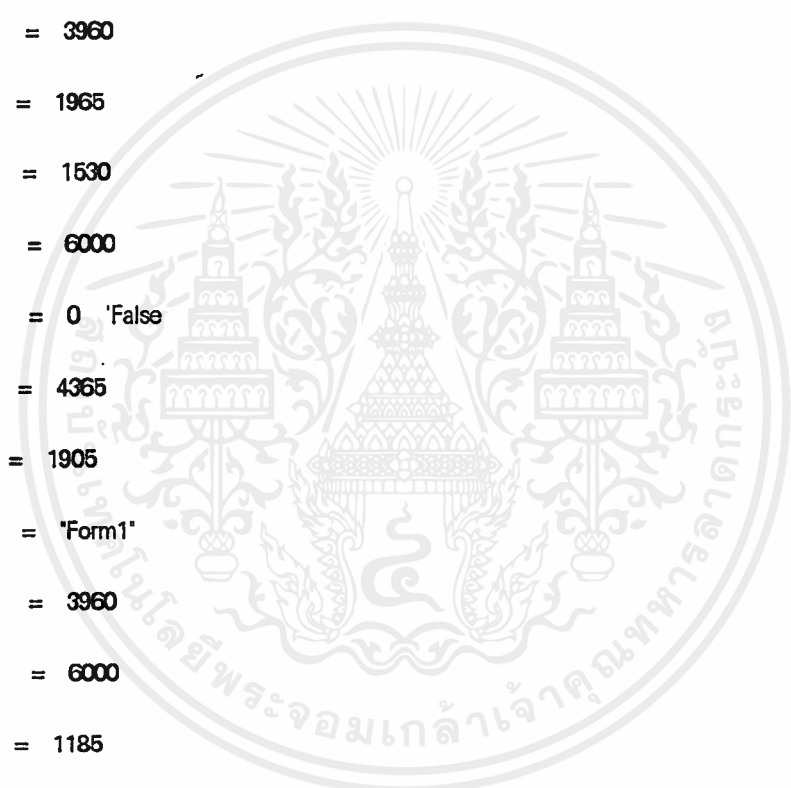
Height = 855

Left = 120

TabIndex = 11

Top = 1200

Width = 4455



Begin TextBox ResultBox

BorderStyle = 0 'None

Enabled = 0 'False

Height = 496

Left = 120

MultiLine = -1 'True

TabIndex = 12

Top = 240

Width = 4215

End

End

Begin Frame frmStatus

Caption = "Status"

Height = 1695

Left = 120

TabIndex = 9

Top = 2160

Width = 4455

Begin TextBox StatusBox

BorderStyle = 0 'None

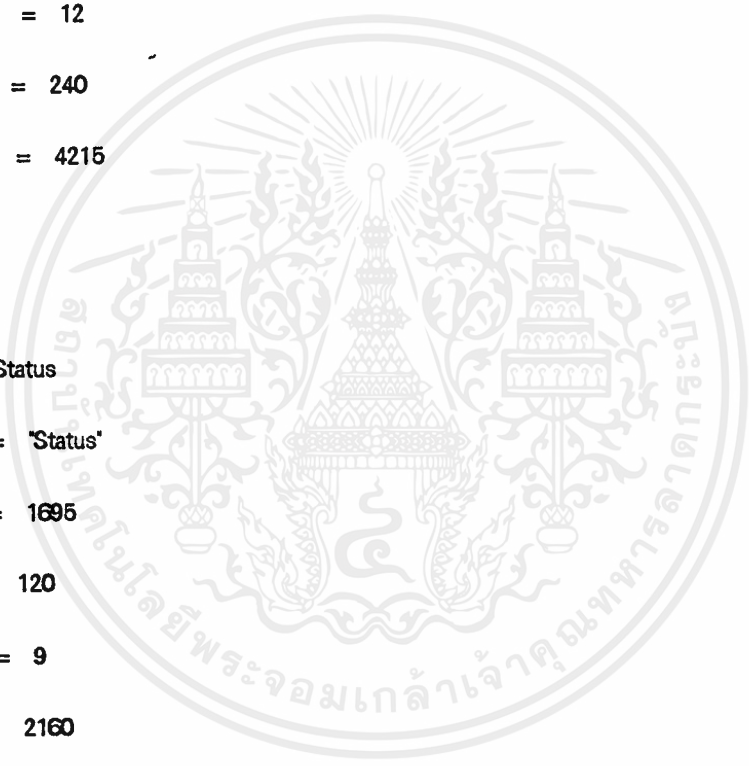
Enabled = 0 'False

Height = 1335

Left = 120

MultiLine = -1 'True

TabIndex = 10



TabStop = 0 'False

Top = 240

Width = 4215

End

End

Begin CommandButton AboutButton

Caption = "&About"

Height = 495

Left = 4680

TabIndex = 6

Top = 120

Width = 1215

End

Begin CommandButton DataOKbutton

Caption = "&OK"

Height = 375

Left = 2880

TabIndex = 2

Top = 720

Width = 1575

End

Begin TextBox UNITInput

Height = 375

Left = 1680

TabIndex = 1

Tag = "Unit"

Top = 720

Width = 1095

End

Begin TextBox PLUinput

Height = 375

Left = 1680

TabIndex = 0

Tag = "PLU"

Top = 240

Width = 2775

End

Begin CommandButton DisconnectButton

Caption = "&Disconnect"

Height = 495

Left = 4680

TabIndex = 4

Top = 1800

Width = 1215

End

Begin CommandButton ConnectButton

Caption = "&Connect"

Height = 495

Left = 4680

TabIndex = 3

Top = 1200

Width = 1215

End

Begin CommandButton ExitButton

Caption = 'E&xit'

Height = -495

Left = 4680

TabIndex = 5

Top = 3360

Width = 1215

End

Begin Shape ConnectStatus

BackStyle = 1 'Opaque

BorderColor = &H00000000&

FillColor = &H000000FF&

FillStyle = 0 'Solid

Height = 735

Left = 240

Top = 240

Width = 615

End

Begin Label Label2

Caption = 'UNIT'

Height = 375

Left = 1080

```

TabIndex = 8
Top = 720
Width = 495

```

End

Begin Label Label1

```

Caption = "PLU"
Height = 375
Left = 1080
TabIndex = 7
Top = 240
Width = 495

```

End

End

' Click the AboutButton button

Sub AboutButton_Click ()

```

frmAbout.Show 1 ' Show the frmAbout form in Modal mode

```

End Sub

! Connect to Database Server using ODBC API

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub Connect_DB ()
```

```
' Return Code
```

```
Dim rc As Integer
```

```
' Change pointer to HourGlass
```

```
frmMain.MousePointer = 11
```

```
rc = ODBC_Connect(frmMain.hWnd, gszDSN, gszSellerID, gszPassword)
```

```
' Back to normal pointer
```

```
frmMain.MousePointer = 0
```

```
If (rc = True) Then
```

```
' Connect successfully
```

```
' *** MsgBox "Connection successful", MB_IconInformation, "Connection Status"
```

```
' Display status
```

```
StatusBox.Text = "Connected"
```

```
' Set connect indicator to green color
```

```
ConnectStatus.FillColor = &HFF00&
```

```
' Toggle off Connect and on Disconnect button
```

```
ConnectButton.Enabled = False
```

```
DisconnectButton.Enabled = True
```

```
Else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

' Connect fail

Beep

' *** MsgBox "Connection failure", MB_IconStop, "Connection Status"

StatusBox.Text = "Connect fail"

End If

End Sub

' Click the ConnectButton button

Sub ConnectButton_Click ()

If (gConnect.fConnected = True) Then

' Already connected

Beep

' *** MsgBox "Already connected to database server", MB_IconExclamation, "Warning"

StatusBox.Text = "Already connected"

Else

' Not yet connected then try connect to database server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Call Connect_DB
```

```
End If
```

```
End Sub
```

```
' Click the DataOKButton button
```

```
Sub DataOKbutton_Click ()
```

```
' Send SQL insert command to server
```

```
' Temporary string
```

```
Dim szTemp As String
```

```
' Generate SQL insert statement
```

```
szTemp = "INSERT INTO selling(plu, sid, posid, time, unit) " + "VALUES ( "
```

```
szTemp = szTemp + "" + UCase$(PLUinputText) + ", "
```

```
szTemp = szTemp + "" + UCase$(gszSellerID) + ", "
```

```
szTemp = szTemp + "" + UCase$(gszPosID) + ", "
```

```
szTemp = szTemp + "" + Format$(Now, 'yyy-mm-dd-hh.mm.ss.000000') + ", "
```

```
szTemp = szTemp + Unitinput.Text
```

```
szTemp = szTemp + ' )'
```

```
' Display status
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
StatusBox.Text = "SQL Statement : " + szTemp
```

```
' Change pointer to HourGlass
```

```
frmMain.MousePointer = 11
```

```
rc = ODBC_Execute(szTemp)
```

```
' Back to normal pointer
```

```
frmMain.MousePointer = 0
```

```
If (rc = True) Then
```

```
    ' *** MsgBox "Insert data OK", MB_IconInformation, "Insert Status"
```

```
    StatusBox.Text = "Insert data OK"
```

```
Else
```

```
    Beep
```

```
    ' *** MsgBox "Insert fail", MB_IconStop, "Insert Status"
```

```
    StatusBox.Text = "Insert data fail"
```

```
End If
```

```
End Sub
```

```
' Disconnect from Database Server using ODBC API
```

```
Sub Disconnect_DB ()
```

```
If (gConnect.fConnected = True) Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
' Change pointer to HourGlass
```

```
frmMain.MousePointer = 11
```

```
rc = ODBC_Disconnect()
```

```
' Back to normal pointer
```

```
frmMain.MousePointer = 0
```

```
If (rc = True) Then
```

```
' Disconnect successfully
```

```
' *** MsgBox "Disconnection successful", MB_IconInformation, "Disconnection Status"
```

```
' Display status
```

```
StatusBox.Text = "Disconnected"
```

```
' Reset connect indicator to red color
```

```
ConnectStatus.FillColor = &HFF&
```

```
' Toggle on Connect and off Disconnect button
```

```
ConnectButton.Enabled = True
```

```
DisconnectButton.Enabled = False
```

```
Else
```

```
' Disconnect fail
```

```
Beep
```

```
' *** MsgBox "Disconnection failure", MB_IconStop, "Disconnect Status"
```

```
StatusBox.Text = "Disconnect fail"
```

End If

End If

End Sub

' Click the DisconnectButton button

Sub DisconnectButton_Click ()

Call Disconnect_DB ' Try disconnect database server

End Sub

' Click the ExitButton button

Sub ExitButton_Click ()

If (MsgBox("Do you want to exit this program now ?", MB_YesNo + MB_IconQuestion, "Exit") = IDYes)

Then

' Signal "unload" to the frmMain form

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Unload frmMain

End If

End Sub

' Load the frmMain form

Sub Form_Load ()

' Load the frmAbout form

Load frmAbout

' Assume not yet connect

gConnect.fConnected = False

ConnectButton.Enabled = True

DisconnectButton.Enabled = False

' ***** Dummy setting up *****

gszDSN = "TBPRJ"

gszSellerID = "ADEK"

gszPosID = "P0001"

gszPassword = ""

' ***** Dummy setting up *****

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

' Connect Database

Call Connect_DB

' Setting up some ODBC parameter

gExecute.IRowsetSize = 15

gExecute.ICursor = SQL_CURSOR_FORWARD_ONLY

gExecute.IConcur = SQL_CONCUR_READ_ONLY

gExecute.IMaxRows = 0

End Sub

' Unload the frmMain form

Sub Form_Unload (Cancel As Integer)

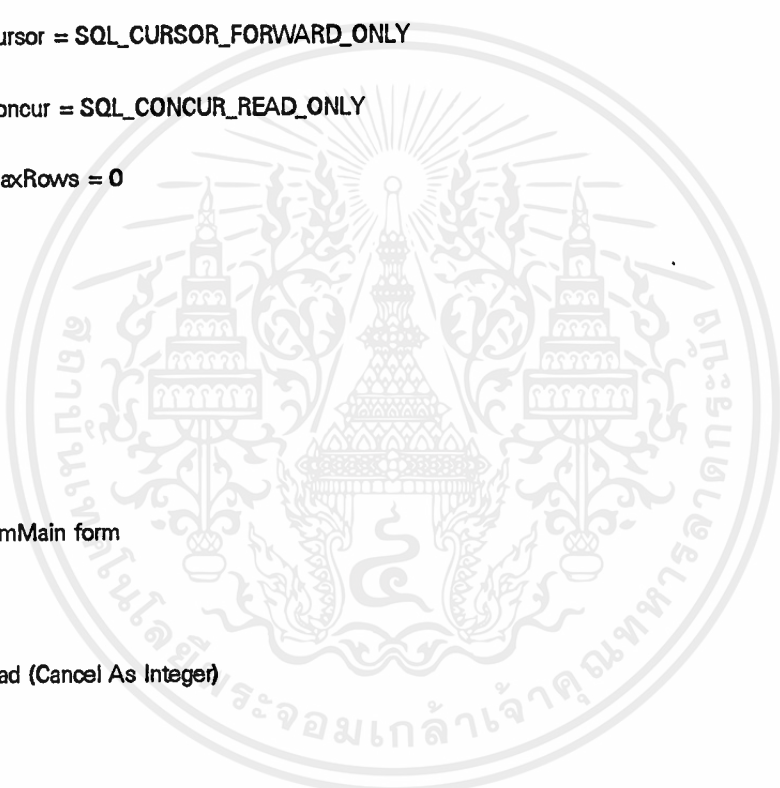
' Disconnect database server

Call Disconnect_DB

' Unload the frmAbout form

Unload frmAbout

End Sub



' Get into PLUinput box

Sub PLUinput_GotFocus ()

' Remove old item

PLUinput.Text = ""

End Sub

' Get into UNITinput box

Sub UNITinput_GotFocus ()

' Remove old item

Unitinput.Text = ""

End Sub



ชื่อแฟ้ม MY_G_VAR.BAS

' Data Source Name (use for ODBC connection)

Global gszDSN As String

' Seller ID (as UserID when login to server)

Global gszSellerID As String

' POS ID

Global gszPosID As String

' Password

Global gszPassword As String



กิตติกรรมประกาศ

ผู้จัดทำขอขอบคุณ ท่านอาจารย์บรรจง ปิยะธำรง เป็นอย่างยิ่งที่ได้ให้คำแนะนำเสมอมา

ขอขอบคุณ ฝ่าย Personnel Software Product (PSP) บริษัท IBM ประเทศไทย จำกัดที่ได้ซื้อโปรแกรม และหนังสือคู่มือ DB2/2 โดยมีคุณเจษฎา คุณวัชรระ เป็นอาทิ

ขอขอบคุณ คุณสุรพัฒน์ บริษัท Interchange ที่ได้ให้แนวคิดในการทำโครงการนี้ และซื้อเนื้อหนังสือ Client/Server with OS/2

ขอขอบคุณ คุณวิโรจน์ และคุณนภาพล บริษัท AR/IP (นิตยสาร Byte ไทยแลนด์) ที่ได้ซื้อเนื้อหนังสือ Inside ODBC

ขอขอบคุณ พี่ป๊อ(ภัทร) ที่ได้ซื้อเนื้อหนังสือ Visual Basic

ขอขอบคุณ น้องตีเล็ก(teelek) น้องหนึ่ง(ten) และน้องเต๋า(tao) และน้องๆที่ได้มาช่วยจัดพิมพ์เอกสาร

ท้ายสุดนี้ขอขอบพระคุณ คุณแม่ คุณพ่อ คุณครูบาอาจารย์ทุกท่าน ที่อุทิศส่ำห้สั่งสอนข้าพเจ้ามาตั้งแต่ อดีตจวบจนปัจจุบัน .

บรรณานุกรม

Robert Orfali and Dan Harkey, 'Client/Server Programming with OS/2 2.0 second edition', Van Nostrand Reinhold, 1111p, 1992

Kyle Geiger, 'Inside ODBC', Microsoft Press, 482p, 1995

IBM, 'DATABASE 2 Installing and Using DB2 Clients for Windows version 2', International Business Machines Corporation, 132p, 1995

Bill Potter, Taylor Maxwell and Bryon Scott, 'Visual Basic Superbible second edition', Waite Group Press, 1620p, 1993

