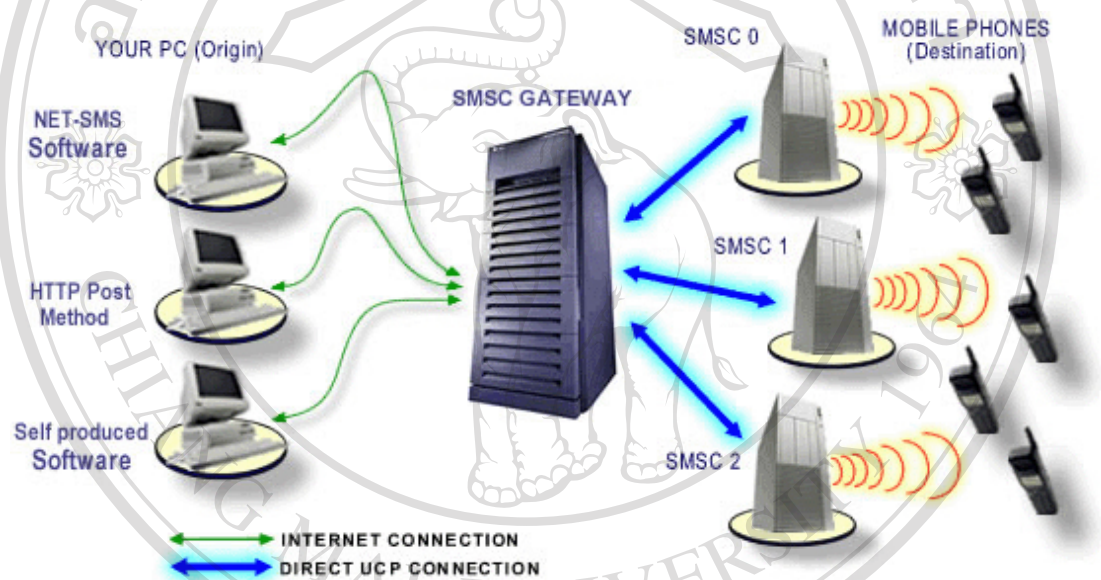


## บทที่ 2

### ทฤษฎีและเอกสารที่เกี่ยวข้อง

ในการศึกษาพัฒนาระบบสำรองข้อมูลและกู้คืนข้อมูลระยะไกล ของเครื่องคอมพิวเตอร์แม่ข่าย ด้วยบริการรับส่งข้อความ ผ่านโทรศัพท์เคลื่อนที่ และอินเทอร์เน็ต นั้นทางผู้ค้นคว้าพบอิสรระพบว่ามีแนวคิดและทฤษฎี ที่นำมาใช้ในการศึกษา ดังรายละเอียดตามลำดับดังนี้

#### 2.1 หลักการทำงานของเอสเอ็มเอส Short Message Service (SMS)



รูป 2.1 ระบบการทำงานของ SMS ของผู้ให้บริการ

(แหล่งที่มา : <http://www.trio4u.com/>)

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่  
Copyright © by Chiang Mai University  
All rights reserved

SMS ย่อมาจากคำว่า Short Message Service หรือเป็นบริการส่งข้อความสั้น ลักษณะการใช้งานจะคล้ายกับการส่งอีเมล แต่จะสามารถส่งข้อความได้ไม่เกิน 160 ตัวอักษรผ่านทางโทรศัพท์มือถือ

### จุดเด่นของบริการ SMS คือ

จากรูป 2.1 เราสามารถส่งเอสเอ็มเอส ไปยังผู้รับโดยไม่ต้องกังวลว่า ณ ขณะนั้นผู้รับปลายทางจะอยู่ในบริเวณที่อับสัญญาณ โทรศัพท์หรือไม่ เนื่องจากผู้ให้บริการโทรศัพท์ จะทำการเก็บข้อมูลไว้จนกว่าปลายทางมีสัญญาณ จากนั้นทางระบบจึงจะทำการส่งข้อมูลไปให้ในทันที นอกจากนี้แล้วเรายังสามารถส่งข้อความเอสเอ็มเอสที่ได้รับมาต่อไปยังหมายเลขอื่นๆ ได้อย่างไม่จำกัดอีกด้วย

### วิวัฒนาการของการส่ง SMS

เป็นที่ทราบกันดีอยู่แล้วว่าประเทศตะวันตกนั้นเป็นผู้พัฒนาโทรศัพท์มือถือขึ้น ฉะนั้นในยุคแรก ๆ ก็จะมีแต่การส่งความเป็นภาษาอังกฤษเท่านั้น แต่ถึงกระนั้นก็ยังมีการคิดค้นวิธีการส่งข้อความรูปแบบใหม่ๆ ไม่ว่าจะเป็นการใช้สัญลักษณ์ต่าง ๆ ที่อยู่ในเครื่อง มาทำเป็นตัวการ์ตูน หน้าคนที่แสดงอารมณ์ต่าง ๆ (Emoticon) และเริ่มมีการใช้ “คำย่อ” เพื่อเป็นการประหยัดเนื้อที่ในการส่ง SMS (SMS Abbreviation) ที่ส่งได้เพียง 160 ตัวอักษร ต่อการส่ง 1 ครั้ง จนเป็นที่นิยมกับผู้ใช้งานมือถือทั่วไป ดังตัวอย่างในตาราง 2.1

จนมาถึงยุคหลัง ๆ ที่โทรศัพท์มือถือได้รับความนิยมขึ้นเรื่อย ๆ สิ่งหนึ่งที่ผู้พัฒนาให้ความสำคัญก็คือทำอย่างไรให้ผู้ใช้งานชาวไทย สามารถส่งข้อความภาษาไทยได้ ในแรกเริ่มการส่งข้อความไปอย่างไม่สะดวกนัก เพราะมีการเรียงลำดับตัวอักษรภาษาไทย เหมือนกับภาษาอังกฤษ ในปุ่มกดปุ่ม (Alpha Numeric) ก็จะมีทั้ง ตัวเลข และ ตัวอักษรภาษาไทย เช่น ปุ่ม 2 ก็จะเป็น ตัว ฉ-ฉ หรือ A-B-C ในภาษาอังกฤษ

จนในที่สุดผู้ให้บริการลำดับที่ 3 ในไทยอย่าง “อoredin” ก็ได้คิดค้นรูปแบบการส่งภาษาไทยที่เรียกว่าสมบูรณ์แบบ และเข้าใจง่ายมาก สำหรับผู้ใช้งานมือถือชาวไทย แนวคิดหลักมาจากการม้วนหัวของตัวอักษรในภาษาไทยนั่นเองแล้ว อoredin ก็ตั้งชื่อบริการนี้ว่า Thai SMS โดย Thai SMS นี้จะถูกบรรจุอยู่ในซิมของอoredin ทุกใบ ดังรูป 2.2

แต่การส่ง SMS เป็นภาษาไทยนั้น 1 ครั้ง จะสามารถส่งได้เพียง 70 ตัวอักษร (รวมสระและเว้นวรรค)

ตาราง 2.1 ตัวอย่างสัญลักษณ์ และคำย่อในการส่ง SMS

ตัวอย่าง สัญลักษณ์	ตัวอย่างคำย่อ
:*) ส่งจูบ	N แทน AND (และ)
@:-) คนผมหยิก	ru ok? แทน Are you okay?
:-D หัวเราะ	x แทน Kiss
;-) ล้อเล่นนะ	2day แทน Today



## 2.2 RS232 DB9 Pin Interface



รูป 2.3 หัวเชื่อมต่อ RS 232 DB9 ตัวเมีย (ซ้าย) และตัวผู้ (ขวา)  
(แหล่งที่มา : <http://www.arcelect.com/rs232.htm>)

จากรูป 2.3 หัวเชื่อมต่อ RS-232 เป็นมาตรฐานการเชื่อมต่อข้อมูลแบบอนุกรม (Serial Communication) ที่มีคนนิยมใช้มากที่สุด กำหนดโดย EIA (Electronics Industry Association) หรือสมาคมผู้ประกอบการอุตสาหกรรมอิเล็กทรอนิกส์ ของประเทศสหรัฐอเมริกา ตั้งแต่ปี ค.ศ.1969 โดยมีจุดเริ่มต้นจากความต้องการที่จะกำหนดมาตรฐานการเชื่อมต่อระหว่าง คอมพิวเตอร์ กับโมเด็มในสมัยนั้น ตัวมาตรฐานจะกำหนดสิ่งที่เกี่ยวข้องกับการเชื่อมต่อนี้ด้วยกันทั้งหมด 4 หัวข้อหลัก ได้แก่

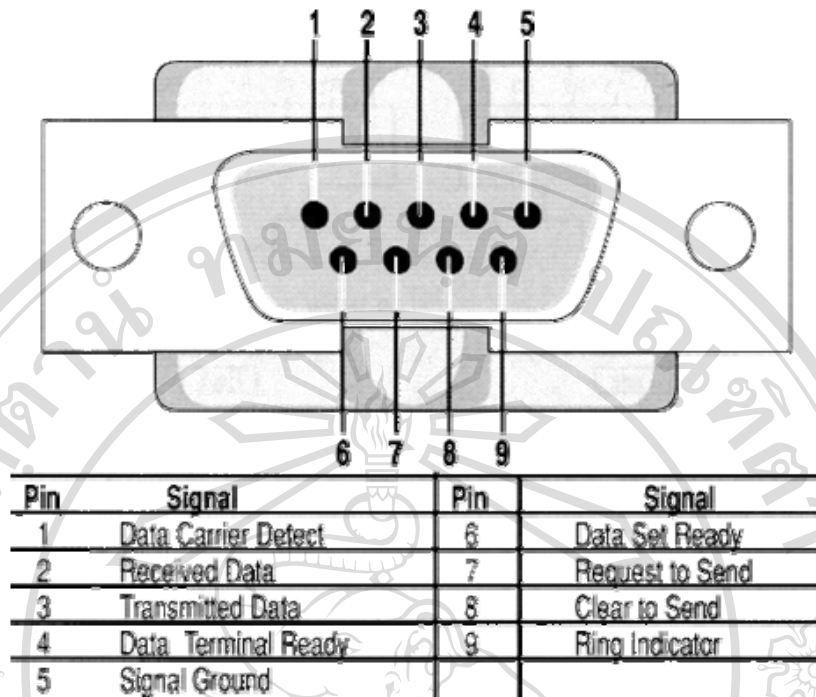
1. คุณสมบัติทางไฟฟ้าของสัญญาณ
2. คุณสมบัติทางกลของการเชื่อมต่อ ซึ่งหมายถึงตัวคอนเน็กเตอร์
3. หน้าที่การทำงานของวงจรสำหรับแลกเปลี่ยนข้อมูล
4. มาตรฐานการเชื่อมต่อสำหรับระบบสื่อสารเฉพาะอย่าง

มาตรฐาน RS-232-C เป็นมาตรฐาน RS-232 ที่มีการปรับปรุงแก้ไขจากมาตรฐานเดิม ได้แก่ RS-232-A และ RS-232-B เทียบกับมาตรฐาน RS-232-D เป็มาตรฐานที่ใหม่กว่า RS-232-C โดยที่มีการเพิ่มข้อกำหนดของคอนเน็กเตอร์แบบ DB เข้าไปด้วย เช่น DB-25 โดยทั่วไปของการเชื่อมต่อข้อมูลแบบอนุกรมตามมาตรฐาน RS-232 คือเป็นการสื่อสารข้อมูลแบบจุดต่อจุด ซึ่งเดิมทีเป็นการสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับ โมเด็ม ซึ่งแท้จริงแล้วทั้งสองฝั่งจะเป็นอะไรก็ได้ การสื่อสารเป็นแบบสองทางพร้อมกัน (Full-duplex) โดยอาจใช้สายสัญญาณอื่นร่วมเพื่อทำแฮนด์เชก (Handshake) หรือไม่ก็ได้ มาตรฐาน RS-232 จำกัดความยาวสายไว้ที่ 50 ฟุต (หรือประมาณ 15 เมตร) สำหรับการส่งสัญญาณที่ความเร็ว 19,200 บิตต่อวินาที โดยที่ความยาวสายจะต้องสั้นลงถ้าต้องการสื่อสารที่ความเร็วสูงขึ้น และถ้ามีสัญญาณรบกวนมาก เช่นในโรงงาน หรือบริเวณใกล้เครื่องจักรที่มีการสวิทช์สัญญาณไฟฟ้ากระแสสูง ก็จะทำให้ต้องมีการลดความเร็วในการส่งสัญญาณลงหรือใช้สายที่สั้นลง เพราะมี สัญญาณรบกวน(Noise)

มาตรฐาน RS-422 หรือ RS-422-A ถูกกำหนดขึ้นโดยสมาคมผู้ประกอบการอุตสาหกรรมอิเล็กทรอนิกส์หรือ EIA เช่นเดียวกับกับมาตรฐาน RS-232 โดยมีจุดมุ่งหมายที่จะแก้ปัญหาเรื่องความยาวของสายสื่อสารโดยใช้การส่ง สัญญาณแบบผลต่าง (Differential) แทนที่จะใช้การส่งสัญญาณแบบอ้างอิงกับจุดกราวด์ (หรือสายดิน) เช่นเดียวกับกับ RS-232 การส่งสัญญาณแบบผลต่างนี้ช่วยลดปัญหาสัญญาณรบกวน (Noise) จาก 2 ปัจจัยด้วยกัน ได้แก่ ปัญหาแรงดันกราวด์ 2 ผังสายไม่เท่ากัน อันเกิดจากกระแสไฟฟ้าที่ไหลในสายกราวด์ที่ยาวมากๆ ก่อให้เกิดความต่างศักย์ (Voltage) และปัญหาสัญญาณรบกวนที่เกิดจากแม่เหล็กไฟฟ้าเหนี่ยวนำในสายได้ โดยหากใช้นำสายไฟที่เคลือบฉนวนวางไว้ใกล้กัน เมื่อมีแรงดันเหนี่ยวนำจะปรากฏแรงดันรบกวนบนสายทั้งสองเท่าๆ กันเป็นผลให้ ตัวรับที่อ่านความต่างศักย์ระหว่างสายอ่านข้อมูลได้เช่นเดิม ทั้งสองปัจจัยนี้เองเป็นสาเหตุที่ทำให้ความต้านทานต่อสัญญาณรบกวนของการสื่อสารแบบ RS-232 ค่อนข้างต่ำกว่า RS-422 ตามมาตรฐาน RS-422 นี้จะใช้สายสัญญาณทั้งหมด 4 เส้น (2 เส้นสำหรับการส่งสัญญาณ และอีก 2 เส้นสำหรับรับสัญญาณ) และสามารถใช้ความยาวสายสัญญาณได้ถึง 4,000 ฟุต (หรือ 1.2 กิโลเมตร) ที่ความเร็ว 100,000 บิตต่อวินาที และการสื่อสารเป็นแบบ 2 ทางพร้อมกัน (Full Duplex)

มาตรฐาน RS-485 กำหนดโดยสมาคมผู้ประกอบการอุตสาหกรรมอิเล็กทรอนิกส์หรือ EIA เป็นมาตรฐานการเชื่อมต่อสัญญาณแบบอนุกรม มีลักษณะการเชื่อมต่อเป็นแบบหลายจุด (Multi-point) หรือ Multi-drop สายสัญญาณที่ใช้มีทั้งแบบที่เป็น 2 สายและแบบที่เป็น 4 สาย การต่อแบบหลายจุดนี้ทำให้สามารถมองสายสัญญาณเป็นบัสสัญญาณได้ (Signal Bus) จำนวนคอมพิวเตอร์หรืออุปกรณ์ที่สามารถอยู่บน RS-485 บัสหนึ่งถูกกำหนดไว้ที่ 32 ตัว ในกรณีที่ต้องการเพิ่มจะต้องมีตัวทวนสัญญาณ (Signal Repeater) หรือใช้ตัวส่ง-รับสัญญาณที่มี อิมพีแดนซ์ หรือ (Impedance) ความต้านทานเสมือน สูงขึ้น ซึ่งเราอาจเพิ่มจำนวนจุดเชื่อมต่อขึ้นได้ถึง 128 จุด ความยาวของสายสัญญาณตามมาตรฐาน RS-485 นี้สามารถยาวได้ถึง 1.2 กิโลเมตร เช่นเดียวกับมาตรฐาน RS-422 แต่การสื่อสารจะเป็นแบบสองทางไม่พร้อมกัน (Half Duplex) มีเพียงคอมพิวเตอร์หรืออุปกรณ์ตัวเดียวเท่านั้นที่สามารถส่งสัญญาณออกได้ ณ เวลาหนึ่งๆ ส่วนที่เหลือจะเป็นผู้รับสัญญาณ หรือผู้ฟัง

หัวเชื่อมต่อ RS 232 ที่ใช้สำหรับสาย Data Link สำหรับโทรศัพท์เข้าสู่เครื่องคอมพิวเตอร์จะมีระบบมาตรฐานการเข้าหัวสายอยู่ดังรูป 2.4 และมีรูปแบบมาตรฐานของการรับส่งข้อมูลของสาย Data Link ดังตารางที่ 2.2



รูป 2.4 รายละเอียดการเข้าหัวสาย Data Link ของโทรศัพท์

(แหล่งที่มา : <http://www.acentech.net/>)

ตาราง 2.2 รูปแบบมาตรฐานของการรับส่งข้อมูลของสาย Data Link จากโทรศัพท์สู่คอมพิวเตอร์

Description	Signal	9-pin DTE	Source DTE or DCE
Carrier Detect	CD	1	from Modem
Receive Data	RD	2	from Modem
Transmit Data	TD	3	from Terminal/Computer
Data Terminal Ready	DTR	4	from Terminal/Computer
Signal Ground	SG	5	from Modem
Data Set Ready	DSR	6	from Modem
Request to Send	RTS	7	from Terminal/Computer
Clear to Send	CTS	8	from Modem
Ring Indicator	RI	9	from Modem

(แหล่งที่มา : [http://www.acentech.net](http://www.acentech.net/))

หมายเหตุ : DTE คือ Data Terminal Equipment ตัวอย่างเช่น PC

DCE คือ Data Circuit Terminating Equipment ตัวอย่างเช่น โทรศัพท์มือถือ

### 2.3 การสื่อสารข้อมูลแบบ MBUS และ FBUS

ในการเชื่อมต่อโทรศัพท์เคลื่อนที่กับอุปกรณ์ Terminal เช่น คอมพิวเตอร์ PC จะมีบัสข้อมูลที่ใช้ในการเชื่อมต่อ 2 อย่างคือ

**MBUS** หรือ Message Bus เป็นการสื่อสารข้อมูลแบบ half-duplex และใช้เชื่อมต่อสัญญาณแบบอนุกรม ใช้เชื่อมต่อโทรศัพท์เคลื่อนที่ เพื่อใช้โทรศัพท์เคลื่อนที่เป็นโมเด็มและใช้รับส่งสัญญาณ

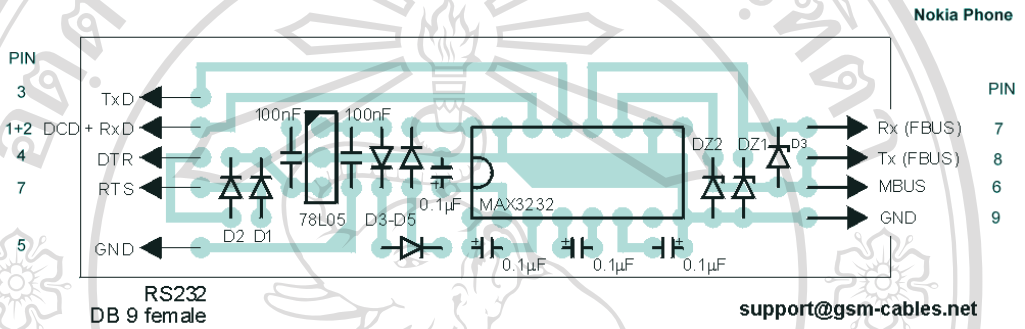
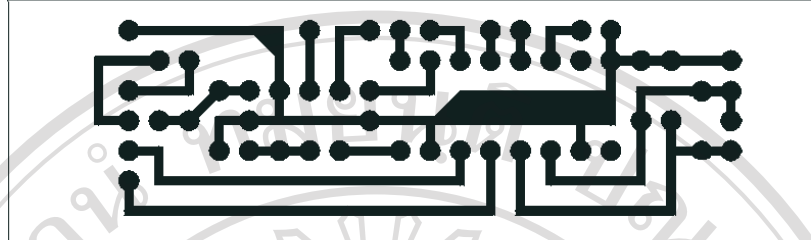
**FBUS** เป็นการสื่อสารข้อมูลแบบ Full-duplex และใช้เชื่อมต่อสัญญาณแบบอนุกรม ครอบคลุมความสามารถการเข้าถึงข้อมูลทุกอย่างที่ MBUS ทำได้ แต่ FBUS จะมีพอร์ตออกมาให้ใช้เพียงโทรศัพท์บางรุ่นเท่านั้น

เนื่องจากสัญญาณที่ออกจาก MBUS และ FBUS ยังไม่เป็นตามมาตรฐาน RS-232 จึงต้องใช้สาย Data Cable เป็นตัวแปลงแรงดันมาตรฐาน โดยสาย Data Cable จะมีคอนเน็คเตอร์เฉพาะโทรศัพท์แต่ละรุ่น ส่วน คอนเน็คเตอร์อีกปลายด้านหนึ่งจะเป็นชนิด DB-9 ไว้สำหรับต่ออุปกรณ์ DTE เช่น PC ดังรูป 2.5 ตัวอย่างรูปแบบวงจรของ Data Cable ที่นิยมใช้ในการเชื่อมโยงแบบ FBUS กับโทรศัพท์มือถือแบบ GSM ได้แก่ SCHEMATICS FOR DAU-9P Cable ดังรูปที่ 2.6 และมีตัวอย่างของ Data Cable โทรศัพท์มือถือ ยี่ห้อ Nokia ได้แก่ Nokia DAU-9P Cable PIN Configuration ดังรูปที่ 2.7



รูป 2.5 ภาพการเชื่อมต่อสายสัญญาณ (Data Link) จากโทรศัพท์มือถือเข้าอุปกรณ์ DTE

GSM CABLES  
<http://gsm-cables.net>



รูป 2.6 รายละเอียด SCHEMATICS FOR DAU-9P Cable การเชื่อมต่อ แบบ FBUS  
 (แหล่งที่มา : <http://gsm-cables.net>)

### FBUS & MBUS Position

1. Charger input voltage
2. Charger control PWM
3. Mic input
4. Signal ground
5. Ear output
6. MBUS 9600 bit/seg
7. FBUS->RX
8. FBUS->TX
9. Ground



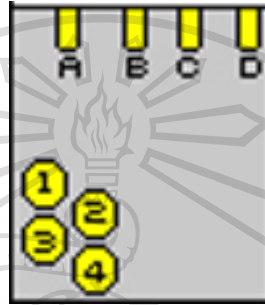
Nokia DAU-9p PIN Configuration

รูป 2.7 รายละเอียด Nokia DAU-9P Cable PIN Configuration

(แหล่งที่มา : <http://www.irpus.or.th/>)

## 2.4 การสื่อสารข้อมูลกับโทรศัพท์ Nokia 3310

จากรูป 2.5 ในสายสัญญาณ (Data Cable) จะมีหัวต่อสัญญาณ (Connector) จากโทรศัพท์มือถือเพื่อเข้าใช้งานได้ โดยหัวต่อสัญญาณของโทรศัพท์มือถือ ยี่ห้อ Nokia รุ่น 3310 เป็นฐานเชื่อมโยกับ แบตเตอรี่ ดังรูป 2.8 มีการทำงานดังตาราง 2.3



รูป 2.8 รายละเอียดการสื่อสารหัวต่อสัญญาณ Phone Connector ของโทรศัพท์รุ่น Nokia 3310  
(แหล่งที่มา : <http://ucables.com/nokia/cables.html>)

ตาราง 2.3 รายละเอียดของ Phone Connector แต่ละหัวสัญญาณ

Pin Number	Pin Name	Direction	Description
1	GND	---	Ground
2	TX	-->	Data Transmission
3	MBUS	<-->	Control bus
4	RX	<--	Data Recive
A	Vbat		Voltage source
B			is battery type Li ion 74.5kohm to ground. NiMh 35Kohm to ground.
C	BSI		battery thermistor. 25K to 40K
D	Gnd	---	Ground

Someone suggested another pinout for Nokia 3310: 1=MBUS; 2=GND; 3=RX; 4=TX

(แหล่งที่มา : <http://ucables.com/nokia/cables.html>)

## 2.5 Linux Crontab

Crontab เป็น Simple Text File ที่ประกอบด้วยรายการคำสั่งที่จะให้รันตามเวลาที่กำหนด โดยคำสั่งดังกล่าวจะสัมพันธ์กับเวลาในการรัน ที่ถูกควบคุมจาก Cron Daemon และถูก Executed ใน System's Background

คำสั่ง crontab จะมีออปชันหลัก 3 ตัวที่ใช้กันเป็นประจำคือ

1. crontab -e สำหรับสร้าง crontab ใหม่หรือแก้ไข
2. crontab -l สำหรับดูรายละเอียด crontab ที่มีอยู่ในระบบของแต่ละ User ของระบบ
3. crontab -r สำหรับลบ crontab (ควรระวังในการใช้งาน)

### รูปแบบการใช้งาน

m	h	dom	mon	dow	command
					+----- คำสั่งที่จะให้ระบบดำเนินการ
				+-----	วันในรอบสัปดาห์ที่จะดำเนินการ 0-6, 0 = วันอาทิตย์
			+-----		เดือนที่จะดำเนินการ 1-12
	+-----				วันในรอบเดือนที่จะให้ทำงาน 1-31
	+-----				ชั่วโมงที่จะให้ระบบทำงาน 0-23
+-----					นาทีที่จะให้ระบบทำงานตามคำสั่ง 0-59

### 2.6 Shell Scripts

หน้าที่พื้นฐานของ Shell โดยทั่วไปนั้นคือการรับคำสั่งจาก User เพื่อทำการบอก OS ให้ Load และ Run Program ต่างๆ อย่างไรก็ตามได้มีการพัฒนาให้ Shell มีความสามารถในการประมวลผลคำสั่งต่างๆ เพื่อให้ทำงานในลักษณะของ Batch ได้ ทั้งนี้โครงสร้างของโปรแกรมหรือ Shell นั้นจะประกอบด้วยตัวแปรระบบ หรือ สภาวะแวดล้อม (Environment) ซึ่งพร้อมที่จะส่งให้โปรแกรมอื่นๆ ประมวลผลต่อ ประโยคเงื่อนไขต่างๆ ระบบ Profile สำหรับวัดเวลาในการประมวลผลของโปรแกรมต่างๆ และ User Defined Function

ในการวัดเวลาที่ใช้ในการทำงานคำสั่งต่างๆ ของ Unix นั้นสามารถทำได้โดยใช้ time นำหน้าคำสั่งที่ต้องการประมวลผล ซึ่งโปรแกรมจะบอก OS ให้ทำการจับเวลา และ รายงานผลเวลาที่ใช้ในการประมวลผลหลังจากทำงานเสร็จ เช่น `time ls -al /bin/*`

สำหรับกรณี Shell บน Unix นั้น การเขียนคำสั่งแต่ละคำสั่งจะต้องจบคำสั่ง ด้วยบรรทัดใหม่ หรือ เครื่องหมาย semi-colon เสมอ ";" และกรณีที่คำสั่งมีความยาวมาก (ไม่จบภายใน 1 บรรทัด) ให้จบบรรทัดด้วยเครื่องหมาย "" แล้วพิมพ์ต่อในบรรทัดถัดไป เช่น

```
echo "Test"; ls;
echo "Hello world"
```

การเขียน Comment ใน Shell Script จะใช้เครื่องหมาย “#” ทั้งนี้ในการเขียนโปรแกรมที่เป็นภาษา Script ต่างๆ บน Unix นั้น ในบรรทัดแรกเราควรระบุชื่อโปรแกรมที่จะทำการประมวลผล Script นั้นๆ ไว้ด้วยเช่น #! /bin/ksh หรือ กรณีเป็น Perl อาจเขียนเป็น #! /usr/bin/perl

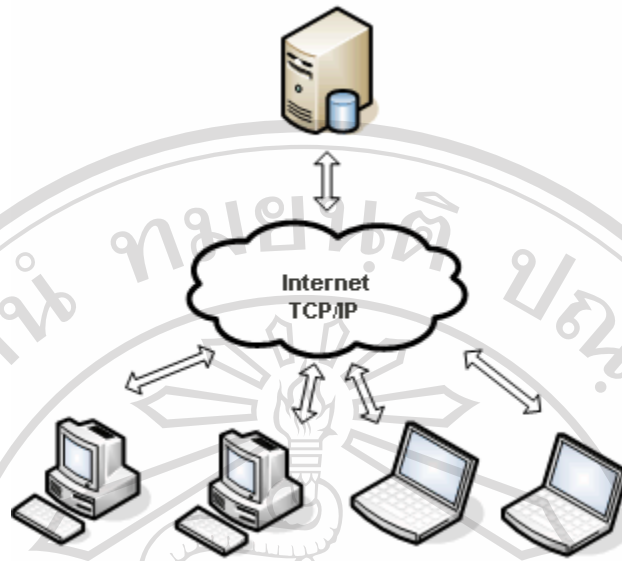
## 2.7 Client-Server Concept

ในยุคแรกๆ ของการพัฒนาซอฟต์แวร์คอมพิวเตอร์นั้น จะใช้รูปแบบการพัฒนาแบบ Centralized Computing กล่าวคือ จะมีคอมพิวเตอร์ขนาดใหญ่ ในระดับ Mini-Computer หรือว่า Mainframe Computer อยู่หนึ่งเครื่อง แล้วก็มี Dumb-Terminal ที่ผู้ใช้งานใช้ในการเชื่อมต่อเข้าไปใช้งานเครื่องคอมพิวเตอร์หลักได้ ซึ่ง Dumb-Terminal นั้น จะมีแค่จอภาพ และ คีย์บอร์ดเท่านั้น ไม่มีส่วนในการประมวลผล ดังนั้นการประมวลผลทั้งหมด จะอยู่ที่เครื่องคอมพิวเตอร์หลัก

เมื่อคอมพิวเตอร์ส่วนบุคคล (Personal Computer) มีการพัฒนาให้ดีขึ้นเรื่อยๆ ก็ทำให้ความนิยมในการใช้งานเครื่อง Mini-Computer หรือว่า Mainframe ลดลงเป็นอย่างมาก จนเครื่องคอมพิวเตอร์ส่วนบุคคลได้เข้าไปแทนที่ในสำนักงานต่าง ๆ โดยในระยะแรก ยังเป็นแบบ Stand-Alone หรือว่า ต่างคนต่างใช้งานอยู่ โดยการแลกเปลี่ยนข้อมูลก็จะใช้ Floppy Disk ในการแลกเปลี่ยนข้อมูล

เมื่อความจำเป็นในการแลกเปลี่ยนข้อมูลมีมากขึ้น ก็เริ่มมีการเชื่อมต่อคอมพิวเตอร์เข้าด้วยกันโดยใช้ LAN (Local Area Network) และต่อมา ก็เป็น Internet ซึ่งการเชื่อมต่อในระยะแรกนั้น ก็ยังเพื่อแลกเปลี่ยนข้อมูลเท่านั้น และต่อมา ก็เป็นการแบ่งปันทรัพยากรในการใช้งานด้วยการประมวลผลนั้น ก็ยังเป็นแบบตัวใครตัวมัน หรืออาจจะเรียกว่าเป็นแบบ Decentralized Computing ก็ได้

และเมื่อความจำเป็นในการประมวลผลที่ซับซ้อนมีมากขึ้นเรื่อยๆ ก็ทำให้มีการพัฒนาเครื่องคอมพิวเตอร์ขนาดใหญ่ ที่เรียกว่า Server ขึ้นมา โดยทั่วไปแล้ว เครื่อง Server ก็จะมีสถาปัตยกรรมเหมือนกับเครื่องคอมพิวเตอร์ส่วนบุคคลโดยทั่วไป แต่ว่าจะมีหน่วยประมวลผลกลางที่ดีกว่า และอาจจะมีมากกว่าหนึ่งหน่วย มีหน่วยความจำที่ใหญ่กว่า และส่วนประกอบอื่น ๆ ที่ดีกว่า ทำให้สามารถทำงานที่ซับซ้อนกว่าเครื่องคอมพิวเตอร์ส่วนบุคคลได้ แต่อย่างไรก็ตาม งานบางอย่างก็ควรจะทำที่เครื่องคอมพิวเตอร์ส่วนบุคคล เช่น การติดต่อกับผู้ใช้ (User Interface) เป็นต้น ซึ่งหลักการนี้เราเรียกว่า Client-Server Architecture ดังรูป 2.9



รูป 2.9 ตัวอย่าง Client-Server Architecture

สถาปัตยกรรมแบบ Client-Server นั้น จะประกอบไปด้วย 2 ส่วนคือ

**Server** เป็นเครื่องคอมพิวเตอร์ที่ทำหน้าที่ประมวลผลในการทำงาน (Business Logic) และการจัดเก็บข้อมูล (Data Storage) ซึ่งโดยทั่วไปแล้ว ผู้ใช้ จะไม่ได้เข้ามายุ่งเกี่ยวกับเครื่องคอมพิวเตอร์เครื่องนี้ แต่จะใช้โปรแกรมที่อยู่ในเครื่อง Client เข้ามาเชื่อมต่อ เพื่อขอให้มีการประมวลผลหรือว่าร้องขอข้อมูลต่าง ๆ เราอาจจะเรียกเซิร์ฟเวอร์เป็นภาษาไทยได้ว่า เครื่องให้บริการ

**Client** เป็นเครื่องคอมพิวเตอร์ส่วนบุคคล หรือว่าในปัจจุบัน อาจจะเป็นเครื่องคอมพิวเตอร์พกพา (Handheld Computer) หรือว่าโทรศัพท์มือถือแบบฉลาด (Smart Phone) ที่สามารถเชื่อมเข้ากับเครื่อง Server เพื่อขอใช้บริการต่าง ๆ ได้ โดยในการทำงานของเครื่อง Client นั้น จะเน้นติดต่อกับผู้ใช้เป็นหลัก

ถึงแม้จะมีการแยกการทำงานค่อนข้างชัดเจน แต่ทั้ง Client และ Server ก็ต้องทำงานร่วมกัน และการเขียนโปรแกรมทั้งสองส่วนของ Client และ Server ก็ต้องทำร่วมกันอย่างระมัดระวังด้วย

## 2.8 Web Application Concept

เมื่อคอมพิวเตอร์มีการพัฒนาการมาจนถึงช่วงปลายของศตวรรษที่ 20 ระบบอินเทอร์เน็ต (Internet) ก็มีประสิทธิภาพมากพอที่จะใช้ในเชิงธุรกิจ อันทำให้เกิดการเจริญเติบโตของธุรกิจ dot-com การทำธุรกิจบนอินเทอร์เน็ตเป็นไปอย่างรวดเร็ว อีกสาเหตุหนึ่งที่ทำให้เกิดการใช้งานอินเทอร์เน็ต และการทำธุรกิจบนอินเทอร์เน็ตมีการพัฒนาอย่างรวดเร็วก็คือการพัฒนาโปรแกรมที่เรียกว่า Web Browser ซึ่งโปรแกรมฝั่ง Client ที่ใช้เชื่อมต่อเข้าไปเรียกข้อมูล หรือว่าขอใช้บริการจาก Web Server ซึ่งเป็นโปรแกรมฝั่ง Server ที่ให้บริการข้อมูลหรือว่าการประมวลผล ซึ่งการเชื่อมต่อจาก Web Browser ไปยัง Web Server นั้น จะใช้โปรโตคอลที่เรียกว่า HTTP (Hyper-Text Transfer Protocol) เป็นข้อตกลงในการแลกเปลี่ยนข้อมูลระหว่าง Web Browser และ Web Server โดยข้อมูลนั้น จะอยู่ในรูปของ HTML (Hyper-Text Markup Language) เป็นส่วนใหญ่ ซึ่ง HTML เป็นภาษาที่ใช้ในการอธิบายวิธีการแสดงผลใน Web Browser ซึ่ง ไม่จำเป็นว่า Web Browser นั้น จะต้องพัฒนาโดยบริษัทใดบริษัทหนึ่ง เนื่องจาก HTML และ HTTP นั้นเป็นโปรโตคอลแบบเปิด (Open Standard) ที่พัฒนาโดย W3C และ IETF ซึ่งเป็นหน่วยงานการที่ออกมาตรฐานเกี่ยวกับอินเทอร์เน็ต ทำให้ Web Browser จากบริษัทหนึ่ง สามารถใช้ร่วมกับ Web Server อีกบริษัทหนึ่งได้

### Web Application Architecture

Web Application เป็นการพัฒนาโปรแกรมที่ใช้ Web Browser เป็นช่องทางติดต่อกับผู้ใช้ และใช้การเขียนโปรแกรมที่ Web Server เพื่อทำการประมวลผล ซึ่งอาจจะกล่าวได้ว่า Web Application เป็น Client-Server Architecture รูปแบบหนึ่ง โดยในการทำงานบน Web Application Architecture นั้น Web Browser จะต้องทำการส่ง Request หรือว่าคำร้องขอไปยัง Web Server ก่อนเสมอ ซึ่งในการส่ง Request นั้นจะต้องอยู่ในรูปของ URL (Uniform Resource Locators) ซึ่งเป็นวิธีมาตรฐานในการอ้างอิงถึงข้อมูล หรือว่า บริการต่าง ๆ บนอินเทอร์เน็ต โดย URL จะอยู่ในรูปของ

**protocol://server\_address/path/resource**

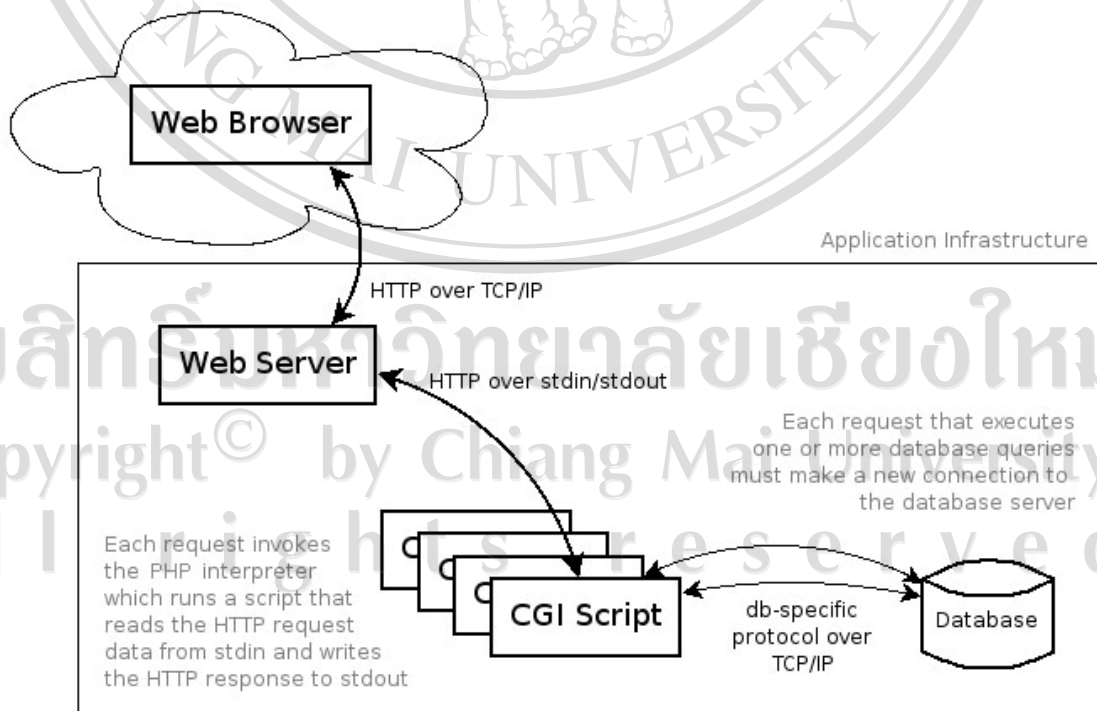
โดย protocol นั้น ถ้าเป็นการร้องขอ Html Page ไปยัง Web Server จะต้องอยู่ในรูปของ http หรือว่า https ส่วน server\_address คือ Internet Address ของ Web Server ที่ให้บริการ ซึ่งอาจจะอยู่ในรูปของ domain name หรือว่า IP Address ก็ได้ ส่วน path ก็คือ directory path ที่ใช้ในการเก็บ resource ดังกล่าว ตัวอย่างเช่น

<http://www.doc.eng.cmu.ac.th/course/cpe207/php/php6-2.php>

จาก URL ดังกล่าวจะได้ว่า

protocol: http  
 server\_address: www.doc.eng.cmu.ac.th  
 path: /course/cpe207/php/  
 resource: php6-2.php

ซึ่ง Web Server ก็จะทำการส่ง Resource หรือว่า Respond มาที่ Web Browser ทำการร้องขอ ซึ่ง Web Server สามารถให้บริการ Resource ได้ในสองลักษณะคือ Static Page และ Dynamic Page ซึ่ง Static Page จะเป็นเอกสารที่ Web Server สามารถส่งกลับไปให้ Web Browser ได้ทันทีโดยไม่ต้องมีการประมวลผลอันใด ส่วน Dynamic Page นั้น เป็นเอกสารที่ต้องมีการประมวลผลก่อนที่จะส่งกลับไป ซึ่งกระทำโดย Web Application นั่นเอง การส่งข้อมูลจาก Web Browser ไปยัง Web Server เพื่อเรียกให้เกิดการประมวลผลนั้น จะกระทำผ่าน CGI (Common Gateway Interface) ซึ่งเป็นวิธีการที่ทำให้ Web Server สามารถเรียกโปรแกรมภายนอกมาประมวลผลให้กับ Web Browser ได้ ดังรูป 2.10



รูป 2.10 Web Application Architecture

### CGI (Common Gateway Interface)

CGI เป็นวิธีการมาตรฐานที่ทำให้ Web Server สามารถเรียกใช้โปรแกรมภายนอกเพื่อทำการประมวลผล Request ที่ส่งมาจาก Web Browser โดย CGI นั้นไม่ขึ้นอยู่กับภาษาใดภาษาหนึ่ง กล่าวคือ Web Server สามารถเรียกใช้ Script ที่เขียนโดยภาษา C, Perl, Java, Python หรือว่า PHP โดย Web Server จะใช้วิธีการดูจาก path ที่เก็บ หรือดูจากนามสกุล เพื่อดูว่าเป็น Script ที่เขียนในภาษาใด และจะได้ทำการประมวลผลได้อย่างเหมาะสม ซึ่ง CGI ยังรวมไปถึงวิธีการส่งข้อมูลจาก Web Browser ซึ่งมักจะผ่านทาง URL หรือว่าผ่าน HTML Form ที่อยู่บน Web page เพื่อนำส่งต่อไปยัง Script อันเป็นวิธีที่ผู้ใช้จะสามารถส่งข้อมูลมายัง Script ได้ ซึ่งการเขียนโปรแกรมผ่าน CGI นั้น มีความพิจารณาอยู่สองสามประการคือ ประการแรกคือ โปรแกรม CGI เป็น Client-Server Application ที่มีการแยกส่วน Presentation และส่วน Logic ออกจากกันอย่างไม่ชัดเจน ประการที่สองคือ Web Application นั้น เป็น Platform Independent ดังนั้น จะคาดหวังว่า Web Browser ที่ผู้ใช้ ใช้งานอยู่จะเหมือนกับ Web Browser ที่เราใช้งานอยู่ไม่ได้ และประการสุดท้ายที่สำคัญมากคือ การเชื่อมต่อระหว่าง Web Browser และ Web Server จะเป็นแบบ Connectionless นั่นคือ เมื่อ Web Browser ต้องการร้องขอ Resource หรือว่าบริการใด ก็จะมีการสร้าง Connection ไปยัง Web Server แล้วก็จะทำการส่ง Request ไปยัง Web Server เมื่อ Web Server ค้นหา Resource เจอแล้ว หรือว่าทำการประมวลผลเสร็จแล้ว ก็จะมีการส่ง Respond กลับมา แล้วก็จะทำการปิด Connection ที่สร้างไว้ทันที นั่นคือ จะไม่มีการสร้าง Connection ทิ้งไว้ระหว่าง Web Browser และ Web Server ซึ่ง จะส่งผลให้ เมื่อมีการสร้าง Connection ใหม่เข้ามายัง Web Server ตัวเดิม จาก Web Browser อันเดิม ก็จะไม่มีความรู้ได้ว่า Web Browser นี้ ได้ทำการเชื่อมต่อมาก่อนหน้านี้แล้ว อันจะทำให้การพัฒนา Web Application นั้นทำได้ยาก เพราะว่าจะต้องทำการตรวจสอบว่าเป็นใครมาจากไหนทุกครั้ง ทำให้ผู้ออกมาตรฐานของ Web Application จึงได้สร้างสิ่งที่เรียกว่า Session และ Cookie โดย Cookie จะเป็นข้อมูลที่ Web Server ทิ้งไว้ใน Web Browser เพื่อเป็นสิ่งระบุว่า เป็นใครมาจากไหน และเมื่อ Web Browser ทำการติดต่อมายัง Web Server อีกครั้ง ก็จะส่ง Cookie นี้มาด้วย เพื่อยืนยันตัวเอง โดยจะนำเอาค่าใน Cookie ไปเปรียบเทียบกับข้อมูล Session ที่มีอยู่บนเซิร์ฟเวอร์ เพื่อหาข้อมูลเก่าที่เคยเก็บไว้ก่อนหน้านี้ใน Session ได้

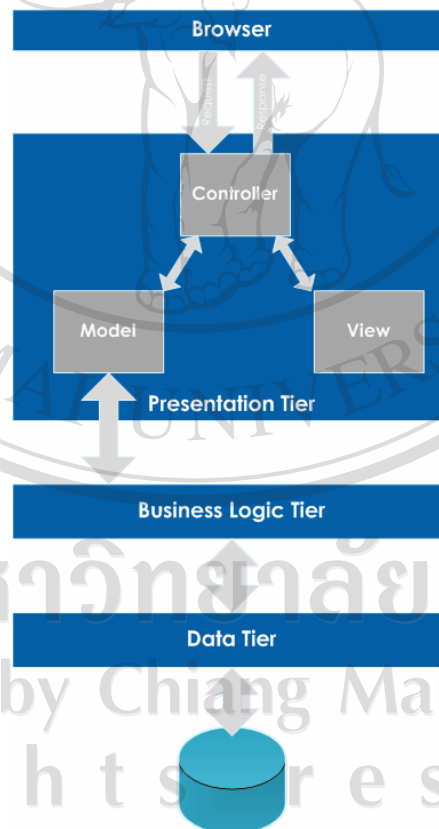
### Three-Tier Application

ในการพัฒนาแอปพลิเคชันสมัยใหม่ จะมีการแยกการทำงานออกจากกันมากขึ้น จากเดิมที่มีเพียง Client-Server ก็มีการขยายเพิ่มออกเป็น สามองค์ประกอบใหญ่ ๆ ซึ่งนำไปสู่แนวคิดของ Three-Tier Architecture ซึ่ง Three-Tier Architecture นั้น จะประกอบไปด้วย

**Presentation Tier** ทำหน้าที่แสดงผลการทำงาน และตอบโต้กับผู้ใช้ โดยจะต้องไม่มีส่วนของการประมวลผล หรือการเก็บข้อมูลใน Tier นี้

**Business Logic Tier** ทำหน้าที่ประมวลผลตามอัลกอริทึมที่กำหนดไว้ ซึ่งผลลัพธ์ของการประมวลผลนั้น จะต้องไม่อยู่ในรูปของ Presentation Format กล่าวคือ จะต้องเป็นข้อมูลล้วน ๆ ไม่มีรูปแบบการแสดงผล มาเกี่ยวข้อง และในส่วนนี้ จะต้องไม่มีการเก็บข้อมูลด้วย

**Data Storage Tier** ทำหน้าที่เก็บข้อมูลเพียงอย่างเดียว โดยข้อมูลที่เก็บ จะไม่มีรูปของการแสดงผล หรือว่าการประมวลผลอยู่ด้วย



ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่  
Copyright© by Chiang Mai University  
All rights reserved

รูป 2.11 Three-Tier Architecture

จากรูป 2.11 จะเห็นว่าแนวคิดแบบ Three-Tier Architecture นั้น จะมีการแยกส่วนการทำงานต่าง ๆ ออกจากกันอย่างชัดเจน และเป็นอิสระต่อกัน การเป็นอิสระต่อกันนั้น หมายถึงว่าเราสามารถเปลี่ยน Presentation Tier ไปเป็นซอฟต์แวร์ หรือว่า Platform อื่นได้ โดยที่ไม่กระทบต่อองค์ประกอบอื่น

ในการพัฒนา Web Application โดยใช้ภาษา PHP นั้น ก็สามารถนำเอาแนวคิดของ Three-Tier Application มาใช้ได้เช่นกัน โดยใช้ PHP จะทำงานในส่วนของ Business Logic ส่วน Web Browser Tier ที่เป็น Client ก็จะทำงานเป็น Presentation Tier และการเก็บข้อมูลนั้น ก็จะเก็บที่ DBMS ที่แยกออกไปต่างหาก เช่น MySQL หรือว่า PostgreSQL เป็นต้น

The logo of Chiang Mai University is a circular emblem. In the center is a detailed illustration of an elephant standing and facing left. Above the elephant's head is a traditional Thai umbrella (parasol). The entire emblem is enclosed within a circular border. The Thai text 'มหาวิทยาลัยเชียงใหม่' is written along the top inner edge of the circle, and 'CHIANG MAI UNIVERSITY 1964' is written along the bottom inner edge. There are decorative floral motifs on the left and right sides of the circle.

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่  
Copyright© by Chiang Mai University  
All rights reserved