

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรมวิเคราะห์ล็อกของบริการเครือข่าย

NETWORK SERVICE LOG ANALYZER



ฉัตรวดี ตักขณาภรณ์

ปิติ ธาคาตั้งสกุล

รฟ.
๐๖3๖1๒
2550

เลขหมู่.....

82037

เลขทะเบียน.....

วัน,เดือน,ปี.....-4 ก.ค. 2551

b..... 11๙ 13๓13
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2550

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมวิเคราะห์ล็อกของบริการเครือข่าย


NETWORK SERVICE LOG ANALYZER

ผู้จัดทำ

1. นายณัฐวุฒิ ลักขณากุล รหัสนักศึกษา 47010243

2. นายปิติ ชาติตั้งสกุล รหัสนักศึกษา 47010452




อาจารย์ที่ปรึกษา
(ผศ.ธนา หงษ์สุวรรณ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมวิเคราะห์ล็อกของบริการเครือข่าย

นายณัฐวุฒิ	ลักษณะกุล	47010243
นายปิติ	ธาดาดังสกุล	47010452
ผศ.ธนา	หงษ์สุวรรณ	อาจารย์ที่ปรึกษา
ปีการศึกษา 2550		

บทคัดย่อ

ปฏิญานิพนธ์ฉบับนี้เสนอรายละเอียดของโปรแกรมวิเคราะห์ล็อกของบริการเครือข่าย ซึ่งโปรแกรมนี้จะทำงานโดยการรวบรวมข้อมูลล็อกเรคคอร์ด ของเซิร์ฟเวอร์และรวมไปถึงอุปกรณ์ในเครือข่าย จากนั้นนำมาคัดกรองและคำนวณเพื่อให้ได้ผลที่ต้องการ นำไปเก็บไว้ในฐานข้อมูล จากนั้นนำมาวิเคราะห์และสรุปเป็นรายงานการทำงานของเซิร์ฟเวอร์ ช่วยให้มองเห็นประสิทธิภาพการทำงานได้มีประสิทธิภาพและง่ายขึ้น โดยรายงานจะสรุปข้อมูลในหลายหลายรูปแบบ เช่น หมายเลขไอพี ระยะเวลาการใช้งาน จำนวนฮิต ฯลฯ ซึ่งข้อมูลที่ได้สะดวกต่อการนำไปใช้งานด้านอื่นต่อเช่น การเก็บค่าบำรุงรักษา, วิเคราะห์ข้อมูลทางธุรกิจ ฯลฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Network Service Log Analyzer

Mr. Nattawoot Luxkanakul 47010243

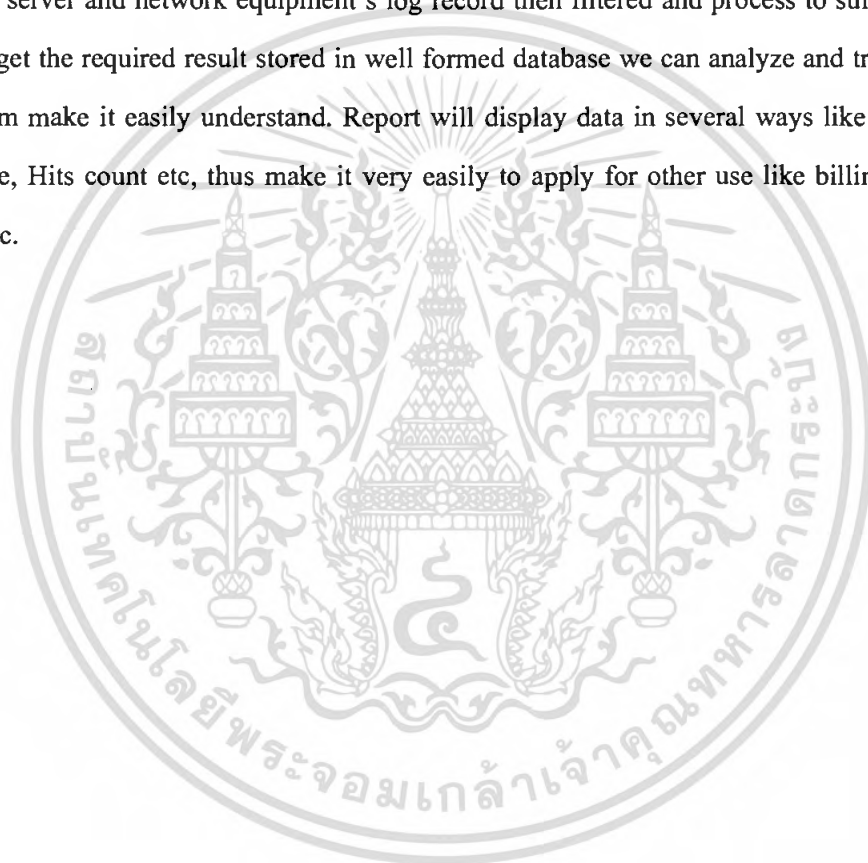
Mr. Piti Thadatungsakul 47010452

Ast. Prof. Mr. Thana Hongsuwun Advisor

Academic Year 2007

Abstract

This thesis proposed Network Service Log Analyzer (NSLA). NSLA working by collecting server and network equipment's log record then filtered and process to suitable result. After we get the required result stored in well formed database we can analyze and translate it to report from make it easily understand. Report will display data in several ways like IP address, time usage, Hits count etc, thus make it very easily to apply for other use like billing, business analyze etc.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอกราบขอบพระคุณ ผศ.ดร.ชนา หงษ์สุวรรณ อาจารย์ผู้ควบคุมปริญญาานิพนธ์ ซึ่งคอยให้คำปรึกษา คำแนะนำมาโดยตลอด รวมถึงความเอาใจใส่ที่อาจารย์มีให้ คอยสละเวลามาให้คำชี้แนะ ให้แนวทางในการทำงาน และทุกๆคำแนะนำดีๆ ข้าพเจ้าซาบซึ้งเป็นอย่างมาก

ขอกราบขอบพระคุณ คณาจารย์ภาควิชาภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกๆท่าน ที่คอยอบรมประสิทธิ์ประสาทวิชาให้ข้าพเจ้า

ขอกราบขอบพระคุณ ภาควิชาภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่คอยให้การสนับสนุน ทั้งด้านสถานที่และอุปกรณ์การทดลอง

ขอบคุณเพื่อน ๆ และพี่ ๆ ในหมู่มิตรสหายสำหรับเสียบงและคำแนะนำดี ๆ ที่มีมาเสมอถึงแม้จะจบออกไปแล้วก็ตาม

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัว ที่ได้อบรมสั่งสอน เลี้ยงดูจนเติบโตใหญ่ รวมถึงความเข้าใจและกำลังใจที่มีให้เสมอมา จนกระทั่งข้าพเจ้าสามารถทำปริญญาานิพนธ์ ฉบับนี้จนสำเร็จ

หวังว่าปริญญาานิพนธ์ฉบับนี้จะก่อให้เกิดประโยชน์แก่ผู้ที่สนใจ คุณค่าที่ได้จากปริญญาานิพนธ์นี้ ข้าพเจ้าขอมอบแด่ผู้มีอุปการคุณทุกๆ ท่าน

ณัฐวุฒิ ลักขณากุล
ปิติ ธาตาดั่งสกุล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	I
1.1 ความสำคัญและที่มาของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 วิธีการดำเนินงาน.....	1
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.6 ส่วนประกอบของโครงการ.....	2
บทที่ 2 ทฤษฎี.....	3
2.1 โพรโตคอล FTP.....	3
2.2 RDBMS.....	10
2.3 Log files	15
2.4 Log File in Apache HTTP Server.....	15
2.5 Log File in Internet Information Services(IIS).....	19
2.6 Log File in Sun™ ONE Web Server(iPlanet).....	22
2.7 WebSphere Application Server (WAS) Log Format	22
2.8 ผลการศึกษาโปรแกรมวิเคราะห์ล็อกไฟล์ที่มีอยู่ในปัจจุบัน.....	23
บทที่ 3 การออกแบบโปรแกรมและการพัฒนา.....	26
3.1 แนวคิดในการออกแบบ.....	26
3.2 ความสามารถของโปรแกรม.....	26
3.3 ภาพรวมของโปรแกรม.....	28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
3.4 รายละเอียดแต่ละส่วนของโปรแกรม	33
3.4.1 Collectors	33
3.4.2 Database	37
3.4.3 Report	39
บทที่ 4 การทดลองและผลการทดลอง.....	41
4.1 การใช้งานในส่วนของ collect.....	41
4.2 การ view Report.....	45
บทที่ 5 บทวิจารณ์และสรุป.....	52
5.1 บทสรุป.....	52
5.2 วิจารณ์สิ่งที่ได้จากโครงงาน.....	52
5.3 ปัญหาอุปสรรคและแนวทางแก้ไข.....	52
5.4 แนวทางการพัฒนาต่อ.....	52
บรรณานุกรม.....	54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงข้อมูลแต่ละบิตในFlag.....	7
2.2 แสดง Fields ที่มีการเข้าถึงเป็น Public.....	8
2.3 แสดงเมทอดที่มีการเข้าถึงเป็น Public.....	9
2.4 แสดงเมทอดที่มีการเข้าถึงเป็น private.....	10
2.5 แสดง Public Constructors.....	10
2.6 แสดง Public Properties.....	11
2.7 แสดง Protected Properties.....	12
2.8 แสดง Public Methods.....	12
2.9 แสดง Protected Methods.....	14
2.10 แสดง Public Events.....	14
2.11 แสดง Explicit Interface Implementations.....	14
2.12 แสดงรายละเอียดของตาราง ข้อมูลของโปรแกรม log analyzer ในปัจจุบัน.....	23
3.1 แสดงรายละเอียดของตารางยูสเคส โค้ดอะแกรมของ โปรแกรม.....	29
3.2 แสดงรายละเอียดของตาราง standard format	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 แสดงการทำงานของ โพรโทคอล FTP.....	4
2.2 แสดงการใช้พอร์ตควบคุมและส่งข้อมูล.....	5
2.3 รูปแบบ FTP Header.....	6
2.4 ตัวอย่าง error log.....	15
2.5 ตัวอย่าง Common Log File Format.....	16
2.6 ตัวอย่าง Combined Log Format.....	17
2.7 ตัวอย่างการสร้าง Multiple Access Log.....	18
2.8 ตัวอย่าง W3C Extend Log File จาก Internet.....	20
2.9 ตัวอย่าง IIS Log File.....	21
2.10 ตัวอย่าง Sun™ ONE Web Server (iPlanet) log forma.....	22
2.11 ตัวอย่างการเก็บบันทึก log ใน field ทั้ง 5 ของ WAS.....	22
3.1 แสดงโครงสร้างโดยรวมของโปรแกรม.....	28
3.2 แสดง Use Case Diagram	29
3.3 แสดง Activity Diagram แสดง process การทำงานของโปรแกรม	30
3.4 แสดงหน้าจอ interface ของโปรแกรม	31
3.5 แสดงโครงสร้างของ collector	33
3.6 แสดงลักษณะการเชื่อมโยงสำหรับการวิเคราะห์จากหลาย data source	34
3.7 แสดง flow chart การทำงานของ collector	36
3.8 แสดงการทำงานของส่วน report	39
4.1 แสดงการเลือกปุ่ม Add.....	41
4.2 แสดงผลหลังจากการคลิกปุ่ม Add.....	42
4.3 แสดงการใช้ปุ่ม delete.....	42
4.4 แสดงผลลัพธ์ หลังจากการกดปุ่ม Delete.....	43
4.5 แสดงการเชื่อมต่อระหว่าง Collector.....	43
4.6 แสดงการกำหนดค่า parameter สำหรับ Collector แต่ละตัว.....	44
4.7 แสดงการเลือกปุ่ม Start.....	44
4.8 แสดงการเลือกปุ่ม View report.....	45
4.9 แสดงผลของ Report ส่วนที่ 1.....	45
4.10 แสดงผลของ Report ส่วนที่ 2.....	46

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.11 แสดงผล report ตามวันในแต่ละเดือน.....	47
4.12 แสดงผล report ตาม OS.....	48
4.13 แสดงผล report ของแต่ละเดือน.....	49
4.14 แสดงผล report ตาม ip ที่เข้ามาใช้.....	50
4.15 แสดงผล report ตาม HTTP Status Code.....	51



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของโครงการ

เนื่องจากในปัจจุบันการติดต่อสื่อสารผ่านระบบเครือข่ายพัฒนาไปอย่างมากเมื่อเทียบกับไม่กี่ปีที่ผ่านมา ส่งผลให้เครื่องมีการติดตั้งเครื่องให้บริการเครือข่ายมากขึ้นเพื่อรองรับความต้องการซึ่งอุปกรณ์ในระบบเครือข่ายเหล่านั้น จะมีการลงบันทึกการทำงานหรือเรียกว่าล็อกไฟล์(log file) เพื่อสามารถนำไปประเมินความสามารถในการทำงานของระบบได้ แต่ในขนาดของระบบที่ใหญ่ขึ้นมากนั้นการจะวิเคราะห์โดยไม่มีเครื่องมือช่วยเป็นไปได้ยาก และตัวล็อกไฟล์เองนั้นก็เข้าใจได้ยากสำหรับผู้ที่ไม่มีความรู้ทางด้านนี้ จึงได้ทำการพัฒนาเครื่องมือที่ใช้วิเคราะห์ล็อกไฟล์เหล่านี้ขึ้นมา

โปรแกรมนี้จะช่วยให้การจัดการกับล็อกไฟล์นั้นทำได้ง่ายขึ้น โดยทำหน้าที่ในการรวบรวมล็อกไฟล์จากอุปกรณ์ในระบบเครือข่าย และนำมาคำนวณเพื่อสรุปเป็นรายงานที่ง่ายต่อการวิเคราะห์มากขึ้น และทำให้ผู้ที่ไม่มีความรู้เฉพาะด้านของระบบสามารถเข้าใจได้

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อสร้าง โปรแกรมที่สามารถวิเคราะห์การเข้าใช้งานเซิร์ฟเวอร์ได้อย่างมีประสิทธิภาพ โดยไม่จำเป็นต้องทำงานบนเซิร์ฟเวอร์โดยตรง
2. เป็นเครื่องมือของผู้ดูแลระบบเพื่อช่วยให้เห็นภาพในการเข้าใช้งานเว็บเซิร์ฟเวอร์

1.3 ขอบเขตของโครงการ

1. โปรแกรมสามารถวิเคราะห์ล็อกไฟล์จากเซิร์ฟเวอร์ที่ต้องการได้
 2. โปรแกรมสามารถรับรองรูปแบบล็อกไฟล์ของเว็บเซิร์ฟเวอร์หลายหลายรูปแบบได้
- ดังต่อไปนี้คือ APACHE และ IIS server

1.4 วิธีการดำเนินการ

1. ศึกษาเนื้อหารายละเอียดต่างๆ ที่จำเป็นต้องใช้-
 - ศึกษาหาข้อมูลและรูปแบบของล็อกไฟล์
 - ศึกษาวิธีการเขียนโปรแกรมเพื่อการดึงข้อมูลผ่านเครือข่าย
 - ศึกษาวิธีการเขียนโปรแกรมเพื่อสร้างและจัดการฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ศึกษาและพัฒนาวิธีการคัดกรองและรวบรวมลีดส์

2. ออกแบบโปรแกรมและส่วนประกอบ

- กำหนดหัวข้อที่รายงานจะแสดงให้ดูเช่น สถิติการเข้าใช้งานแยกตามวัน, เพจที่มีฮิตสูงสุด ฯลฯ

- ออกแบบขั้นตอนการทำงานของโปรแกรม

- ออกแบบรูปแบบมาตรฐานที่จะเป็นมาตรฐานสำหรับลีดส์จากหลากหลายฟอร์แมต

- ออกแบบฐานข้อมูลเพื่อที่จะเก็บข้อมูลที่คัดกรองและคำนวณแล้ว

3. ทำการพัฒนาโปรแกรมและส่วนประกอบต่าง ๆ ตามที่ได้ออกแบบไว้

4. ทดลองทำการใช้งานประเมินผลและแก้ไขโปรแกรมและสรุปผลการทำงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้ความเข้าใจเกี่ยวกับการออกแบบแอปพลิเคชันด้วยโปรแกรม C#
2. ได้รับความรู้ความเข้าใจในการเขียนโปรแกรมเกี่ยวกับ Network Programming

1.6 ส่วนประกอบของรายงาน

รายงานฉบับนี้ได้แบ่งเนื้อหาออกเป็น

บทที่ 1 กล่าวถึงความสำคัญและที่มาของโครงงาน วัตถุประสงค์ของโครงงาน ขอบเขตของโครงงาน วิธีการดำเนินงาน ประโยชน์ที่คาดว่าจะได้รับ และส่วนประกอบของรายงาน

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐาน ที่เป็นทฤษฎีที่ศึกษามาใช้ในการออกแบบและพัฒนาโปรแกรม

บทที่ 3 กล่าวถึงการออกแบบ ซึ่งจะกล่าวถึงภาพรวมของโปรแกรม

บทที่ 4 กล่าวถึงการทดลองและผลการทดลอง

บทที่ 5 เป็นส่วนของการสรุปและทดลองรวมไปถึงข้อเสนอแนะ

บทที่ 2

ทฤษฎี

2.1 โพรโทคอล FTP

ในโลกธุรกิจปัจจุบันที่มีการแข่งขันกันมากในทุกๆ ด้าน ไม่ว่าจะเป็นด้านการผลิต ด้านการตลาด การบริหาร การจัดการ และด้านที่ขาดไม่ได้ด้วยเช่นกัน คือ ด้านการสื่อสาร ซึ่งแต่ละธุรกิจมีความจำเป็นต้องใช้ ไม่ว่าจะเป็นการติดต่อสื่อสารกับลูกค้า การติดต่อสื่อสารภายในหรือระหว่างบริษัท ซึ่ง FTP มีส่วนช่วยอย่างมากในการสื่อสารต่างๆ FTP จะช่วยให้การถ่ายโอนข้อมูลเป็นไปได้ง่ายมากขึ้น แต่ละบริษัทหรือหน่วยงานสามารถมีข้อมูลมากมายหลายรูปแบบที่ต้องการสื่อสารไปยังแหล่งอื่น หรือแม้แต่ต้องการข้อมูลที่เป็นประโยชน์จากแหล่งอื่นเข้ามาใช้ เช่น ข้อมูลข่าวสารประจำวัน บทความ ข้อมูลทางสถิติ ผลการทดลองทางวิทยาศาสตร์ เป็นต้น การจะเดินทางไปเอาข้อมูลต่างๆ เอง ก็ถือเป็นความเสี่ยงเวลาโดยเปล่าประโยชน์ ในเมื่อมีเทคโนโลยีเข้ามาช่วยเหลือแล้ว FTP จะเป็นตัวช่วยให้การได้รับข้อมูลเหล่านี้สามารถทำได้ง่ายยิ่งขึ้นเพียงอยู่บนหน้าจอคอมพิวเตอร์เท่านั้น ผู้ใช้งานสามารถใช้ FTP ในการโอนข้อมูลจำนวนมากจากแหล่งที่อนุญาตให้ใช้ได้ ซึ่งเรียกว่าเป็นแหล่งบริการ FTP ซึ่งมักเป็นศูนย์รวบรวมข้อมูลข่าวสารต่างๆ อยู่มาก และเปิดบริการทั่วไป เพียงแค่ผู้ใช้อินเทอร์เน็ตเข้าไปใช้บริการคัดลอกเพิ่มข้อมูลต่างๆ มาใช้งาน

FTP หรือ File Transfer Protocol เป็นโพรโทคอลที่ใช้ส่งผ่านข้อมูลระหว่างเครื่องคอมพิวเตอร์ โดยทำงานบน TCP/IP โพรโทคอล หลักการทำงานของ FTP นั้นจะประกอบไปด้วยคอมพิวเตอร์สองเครื่อง คือ เซิร์ฟเวอร์ และ ไคลเอน โดยทางฝั่ง เซิร์ฟเวอร์ จะต้องรันโปรแกรมที่ทำหน้าที่เป็น FTP เซิร์ฟเวอร์ ซึ่งจะมีหน้าที่รองรับการร้องขอข้อมูลจากเครื่องคอมพิวเตอร์อื่น ๆ ส่วน ไคลเอน นั้นจะมี FTP application เพื่อทำหน้าที่ในการติดต่อสื่อสารกับ เซิร์ฟเวอร์ เมื่อเครื่องคอมพิวเตอร์ทั้งสองสามารถเชื่อมต่อกันได้แล้ว เครื่อง ไคลเอน สามารถจัดการกับไฟล์ที่ ไคลเอน มีสิทธิ์ตามที่ เซิร์ฟเวอร์ อนุญาตได้หลากหลายรูปแบบ เช่น คำนวณโหลด, เปลี่ยนแปลงชื่อหรือลบไฟล์ และอัปโหลดไฟล์บนเครื่อง เซิร์ฟเวอร์ โดยพื้นฐานแล้วเครื่องคอมพิวเตอร์ส่วนใหญ่จะรองรับ FTP Protocol ซึ่งจะส่งผลให้เครื่องคอมพิวเตอร์ที่ใช้ OS คนละตัวยังสามารถจัดการไฟล์ระหว่างกันและกันได้ผ่านโพรโทคอล FTP ซึ่งโพรโทคอลนี้นั้นสามารถที่จะคิดค้นร่วมไปกับเซิร์ฟเวอร์ชนิดอื่น ๆ เพื่อการทำงานที่สมบูรณ์ได้ เช่น Internet host, game servers, voice servers

ข้อมูลของ FTP ที่สื่อสารระหว่างกันมี 2 ประเภทคือ

- ข้อมูล(Data) หมายถึงข้อมูลต่างๆที่ต้องการรับส่ง รวมทั้งไฟล์ที่รับมาจากเซิร์ฟเวอร์ หรือส่งมาจาก ไคลเอนต์แล้วไปเก็บที่เซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ข้อมูลคำสั่ง (Command) FTP จะมีคำสั่งที่ใช้สั่งงานต่างๆ เช่น dir เป็นคำสั่งที่ใช้แสดงชื่อไฟล์หรือไดเรกทอรีในเครื่องเซิร์ฟเวอร์ หรือ get ใช้โหลดไฟล์มาที่เครื่องไคลเอนต์ผ่านโปรแกรม FTP แล้วโปรแกรมจะส่งคำสั่งไปยังเซิร์ฟเวอร์เพื่อทำงาน และแจ้งผลการทำงานกลับมายังไคลเอนต์ ซึ่งผลการทำงานนี้จะนำหน้าด้วยตัวเลข 3 หลัก เป็นรหัสที่ใช้แสดงสถานะการทำงานภายในของ FTP และต่อด้วยข้อความที่เป็นเท็กซ์ต่อท้าย ซึ่งก็คือผลการทำงานหรือคำอธิบายต่างๆ โดยที่ FTP มีกระบวนการภายในที่จะตรวจสอบได้ว่าข้อมูลที่จะรับส่งนี้เป็นประเภทคำสั่งไม่ใช่ตัวข้อมูลที่ต้องการจะโอนย้าย

การที่ FTP สามารถแยกแยะข้อมูลจริงออกจากข้อมูลที่เป็นคำสั่งได้นั้น ถือว่าเป็นหน้าที่การทำงานของโมดูลใน FTP ที่เรียกว่าโปรโตคอล (Protocol Interpreter Module หรือ PI) ซึ่งทำหน้าที่รองรับการทำงานคำสั่งต่างๆของ FTP และในส่วนของข้อมูลที่รับส่งนั้นจะเป็นหน้าที่ของโมดูลโอนข้อมูล (Data Transfer หรือ DT) ซึ่งโมดูลทั้งสองนี้จะต้องทำงานอยู่ทั้งในเครื่องที่เป็นเซิร์ฟเวอร์ และไคลเอนต์



รูปที่ 2.1 แสดงการทำงานของ โปรโตคอล FTP

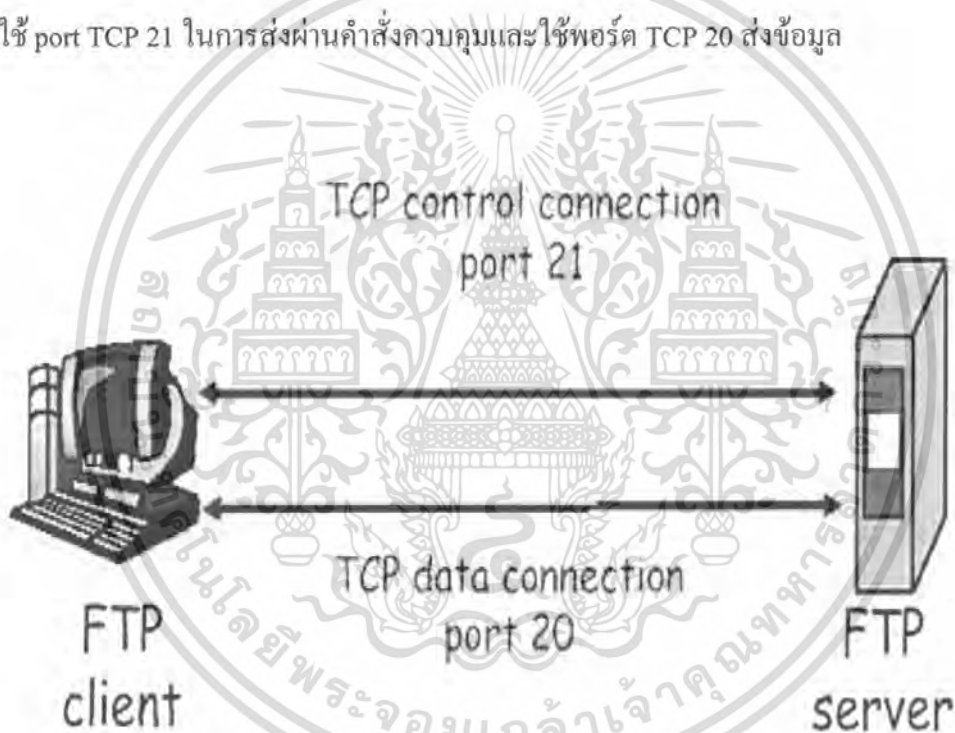
จากรูปแสดงถึงองค์ประกอบและกลไกการทำงานของโปรโตคอล FTP จะเริ่มจากผู้ใช้ (USER) เรียกใช้โปรแกรมผ่าน User Interface และ เมื่อเป็น โปรแกรม FTP พร้อมใช้งานแล้ว ถ้ามีการใช้คำสั่งต่างๆของFTPจะเป็นหน้าที่ของ PI (Protocol Interpreter module) ทำหน้าที่แปลคำสั่งและทำงานตามคำสั่ง ในกรณีที่มีการส่งรับข้อมูลก็จะเป็นหน้าที่ของ DT (Data Transfer module) ซึ่ง โมดูล PI และDTนี้จะอยู่ทั้งด้านของไคลเอนต์และเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประเภทของการล็อกอินในบริการ FTP

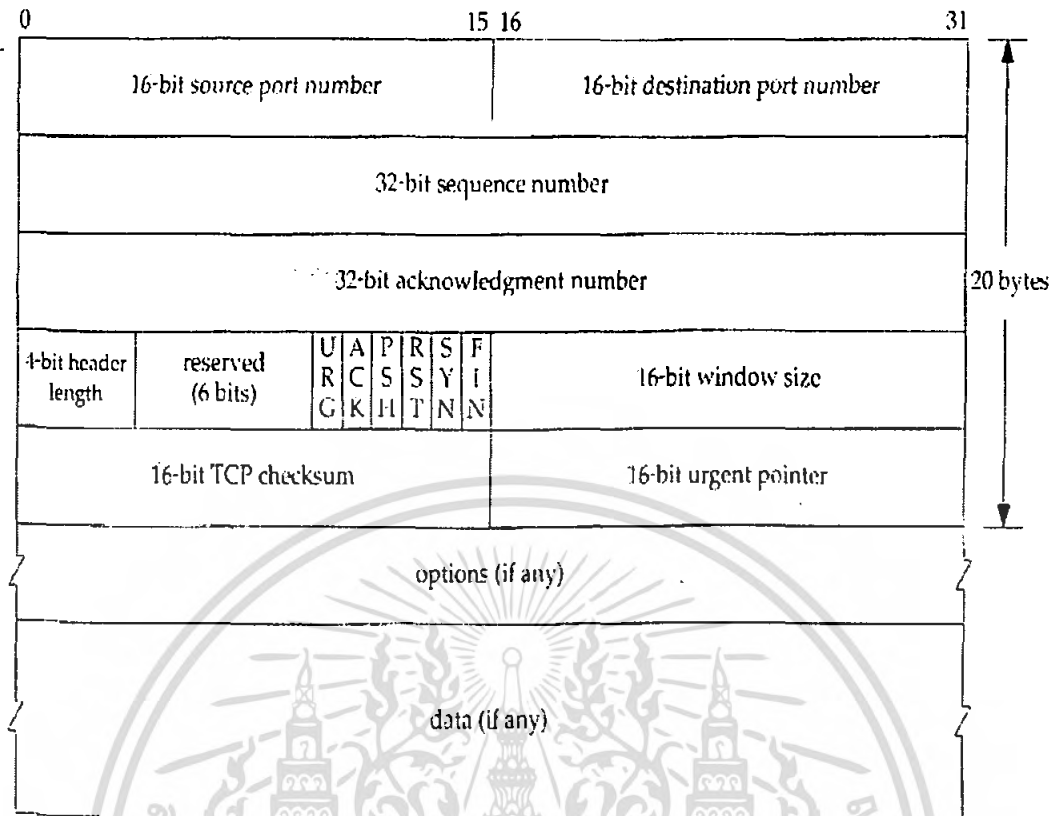
- ล็อกอินด้วยผู้ใช้ที่มีอยู่ในระบบ (Real FTP) ผู้ใช้บริการจะต้องมีบัญชีผู้ใช้อยู่จริงบนเซิร์ฟเวอร์ สามารถเปลี่ยนไคลเอนต์ไปที่อื่นได้
- ล็อกอินด้วยผู้ใช้ที่มีอยู่ในระบบแต่จำกัดขอบเขต (Guest FTP) คล้ายกับ Real FTP ต่างตรงที่ ไม่สามารถเปลี่ยนไคลเอนต์ไปไหนได้เกินขอบเขตที่เซิร์ฟเวอร์กำหนด
- ล็อกอินด้วยผู้ใช้ที่ไม่มีอยู่ระบบ (Anonymous FTP) การบริการ FTP แบบที่เปิดเสรีให้คนทั่วโลกมาใช้บริการ คงเป็นไปได้ที่จะมานั่งสร้างบัญชีผู้ใช้ให้รองรับคนทั่วโลก แบบนี้จึงกำหนดให้ล็อกอินโดยใช้ชื่อ anonymous ส่วนรหัสผ่าน E-Mail Address

FTP ใช้ port TCP 21 ในการส่งผ่านคำสั่งควบคุมและใช้พอร์ต TCP 20 ส่งข้อมูล



รูปที่ 2.2 แสดงการใช้พอร์ตควบคุมและส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 รูปแบบ FTP Header

จากรูปที่ 2.3 มีรายละเอียดดังนี้

- Source Port Number : หมายเลขพอร์ตต้นทางที่ส่งดาต้าแกรม
- Destination Port Number : หมายเลขพอร์ตปลายทางที่จะเป็นผู้รับดาต้าแกรม
- Sequence Number : ฟิลด์ที่ระบุหมายเลขลำดับอ้างอิงในการสื่อสารข้อมูลแต่ละครั้ง เพื่อใช้ในการแยกแยะว่าเป็นข้อมูลของชุดใด และนำมาจัดลำดับได้ถูกต้อง
- Acknowledgment Number : ทำหน้าที่เช่นเดียวกับ Sequence Number แต่จะใช้ในการตอบรับ
- Header Length : โดยปกติความยาวของเฮดเดอร์ TCP จะมีความยาว 20 ไบต์ แต่อาจจะมีมากกว่านั้น ถ้ามีข้อมูลในฟิลด์ option แต่ต้องไม่เกิน 60 ไบต์
- Flag : เป็นข้อมูลระดับบิตที่อยู่ในเฮดเดอร์ TCP โดยใช้เป็นตัวบอกคุณสมบัติของแพ็กเก็ต TCP ขณะนั้นๆ และใช้เป็นตัวควบคุมจังหวะการรับส่งข้อมูลด้วย ซึ่ง Flag มีอยู่ทั้งหมด 6 บิต แบ่งได้ตามตารางที่ 2.1 ดังนี้

ตารางที่ 2.1 แสดงข้อมูลแต่ละบิตใน Flag

Type	Description
URG	ใช้บอกความหมายว่าเป็นข้อมูลด่วนและมีข้อมูลพิเศษมาด้วย(อยู่ใน Urgent pointer)
ACK	แสดงว่าข้อมูลในฟิลด์ Acknowledge Number นำมาใช้งาน
DSH	Gเป็นการแจ้งให้ผู้รับข้อมูลทราบว่าควรส่งข้อมูล Segment นี้ไปยัง Application ที่กำลังรออยู่โดยเร็ว
RST	ยกเลิกการติดต่อ (reset) เนื่องจากในกรณีที่เกิดการสับสนขึ้นด้วยเหตุผลต่างๆเช่น โสตค์มีปัญหา ให้เริ่มต้นสื่อสารกันใหม่
SYN	ใช้ในการเริ่มต้นขอติดต่อกับปลายทาง
FIN	ใช้ส่งเพื่อแจ้งให้ปลายทางทราบว่ายุติการติดต่อ

คำสั่งทั่วไปที่ใช้ได้บน FTP เซิร์ฟเวอร์ เช่น

- ascii - กำหนดให้การรับส่งข้อมูลเป็นรหัส ASCII
- binary - กำหนดให้การรับส่งข้อมูลเป็นเลขฐานสอง
- bye - ออกจาก FTP
- cd - เปลี่ยนไดเรกทอรี
- delete - ลบไฟล์
- dir - แสดงไดเรกทอรี
- get - รับไฟล์จาก remote host เข้ามาเก็บใน local host
- put - ส่งไฟล์จาก local host ไปที่ remote host
- hash - ได้แสดงเครื่องหมาย # หลังจากส่งข้อมูลแต่ละ block เสร็จ

สำหรับคำสั่งของ FTP นี้อาจมีไม่เหมือนกันบนแต่ละ FTP เซิร์ฟเวอร์ ตัวอย่างที่แสดงข้างบนเป็นเพียงคำสั่งพื้นฐานที่น่าจะมีบนทุก FTP เซิร์ฟเวอร์

2.1.1 FTP Web Request Class

เป็น Class ที่ Microsoft .NET พัฒนาขึ้นเพื่อใช้ในการสร้าง FTP Client สำหรับการเขียนโปรแกรมที่ต้องการติดต่อกับ FTP เซิร์ฟเวอร์

ในการเริ่มต้นสร้าง Class นี้ขึ้นมาใช้งานเริ่มต้นจาก method create และระบุ URI ที่ต้องการลงไปเพื่อสร้างการเชื่อมต่อที่สามารถโปรแกรมได้กับ FTP เซิร์ฟเวอร์ โดยเราจะระบุ Username และ password ที่จำเป็นต้องใช้การ login ใน Credential property เมื่อสามารถเชื่อมต่อได้แล้วเราสามารถควบคุมคำสั่งที่ต้องการส่งไปหา sever ผ่านทาง property ที่ชื่อว่า Method

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราต้องการจะส่งไฟล์ขึ้นไปที่ เซิร์ฟเวอร์ ผู้ใช้ต้องเขียนข้อมูลของไฟล์ที่ต้องการไปที่ request stream โดยการเรียกใช้ GetRequestStream method โดยที่ผู้ใช้จะต้องเขียน และปิด stream ให้เรียบร้อยก่อนที่จะส่งไฟล์

การร้องขอติดต่อกับเซิร์ฟเวอร์ทำได้โดยการเรียก GetResponse method ซึ่งเมื่อทำงานเรียบร้อยแล้วจะส่งค่ากลับมาเป็น FtpWebResponse ที่แสดงสถานะการทำงานและ ข้อมูลที่เรา คำนวณโหลคมาจาก เซิร์ฟเวอร์

ผู้ใช้สามารถกำหนด time-out สำหรับการอ่านหรือเขียนไฟล์ไปที่ เซิร์ฟเวอร์ โดยการ กำหนด property ที่ชื่อ ReadWriteTimeout ถ้าช่วงเวลา time-out หหมดลงแล้ว method ที่ติดต่อไปหา เซิร์ฟเวอร์ จะส่ง WebException กลับมาโดยมี status เป็น Timeout

เมื่อทำการคำนวณโหลคไฟล์จาก FTP เซิร์ฟเวอร์ เมื่อทำคำสั่งสำเร็จแล้วไฟล์ที่เราร้องขอไป จะ อยู่ใน response อย่างที่ได้กล่าวมาข้างต้นโดยใช้ GetResponseStream method

โดยยังมี method อื่น ๆ ที่ยังไม่ได้อีกถึงอีกคงที่จะแสดงให้เห็นดังต่อไปนี้

ตารางที่ 2.2 แสดง Fields ที่มีการเข้าถึงเป็น Public

Name	Description
<u>AppendFile</u>	ใช้แทนคำสั่ง APEE ของโปรโตคอล FTP ที่ใช้เพื่อ ระบุไฟล์ที่ตั้งอยู่บน FTP เซิร์ฟเวอร์
<u>DeleteFile</u>	ใช้แทนคำสั่ง DELE ของโปรโตคอล FTP เพื่อใช้ในการลบไฟล์ที่อยู่บน FTP เซิร์ฟเวอร์
<u>DownloadFile</u>	ใช้แทนคำสั่ง RETR ของโปรโตคอล FTP ที่ใช้ในการคำนวณโหลคไฟล์จากเซิร์ฟเวอร์นั้น
<u>GetDateTimestamp</u>	ใช้ในการเรียกดูเวลาของไฟล์ในเซิร์ฟเวอร์
<u>GetFileSize</u>	ใช้แทนคำสั่ง SIZE ของโปรโตคอล FTP ที่ใช้ในการดูขนาดของไฟล์บนเครื่องเซิร์ฟเวอร์
<u>ListDirectory</u>	ใช้แทนคำสั่ง NLIST ของโปรโตคอล FTP ที่ใช้เรียกดูชื่อไฟล์ทั้งหมดในไดเรกทอรี นั้น ๆ บน เซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 (ต่อ)

<u>ListDirectoryDetails</u>	ใช้แทนคำสั่ง LIST ของโปรโตคอล FTP ที่ใช้เรียกดูชื่อไฟล์และรายละเอียดทั้งหมดในไดเรกทอรีนั้น ๆ บนเซิร์ฟเวอร์
<u>MakeDirectory</u>	ใช้แทนคำสั่ง MKD ของโปรโตคอล FTP ที่ใช้ในการสร้างไดเรกทอรีขึ้นมาบนเครื่องเซิร์ฟเวอร์
<u>PrintWorkingDirectory</u>	ใช้แทนคำสั่ง PWD ของโปรโตคอล FTP ที่ใช้ในการแสดงไดเรกทอรีที่ทำงานอยู่ปัจจุบันบนเครื่องเซิร์ฟเวอร์
<u>RemoveDirectory</u>	ใช้แทนคำสั่ง RMD ของโปรโตคอล FTP ที่ใช้ในการลบไดเรกทอรีบนเครื่องเซิร์ฟเวอร์
<u>Rename</u>	ใช้แทนคำสั่ง RENAME ของโปรโตคอล FTP ที่ใช้ในการเปลี่ยนชื่อไดเรกทอรีบนเครื่องเซิร์ฟเวอร์
<u>UploadFile</u>	ใช้แทนคำสั่ง STOR ของโปรโตคอล FTP ที่ใช้ในการอัปโหลดไฟล์ขึ้นไปบนเครื่องเซิร์ฟเวอร์
<u>UploadFileWithUniqueName</u>	ใช้แทนคำสั่ง STORU ของโปรโตคอล FTP ที่ใช้ในการอัปโหลดไฟล์ที่มีชื่อเฉพาะขึ้นไปบนเครื่องเซิร์ฟเวอร์

ตารางที่ 2.3 แสดงเมทอดที่มีการเข้าถึงเป็น public

Name	Description
Equals	ใช้ตรวจสอบดูว่าออบเจกต์ทั้งสองมีค่าเท่ากันหรือไม่
GetHashCode	ใช้เรียก GetHashCode สำหรับการทำฟังก์ชัน hash .
GetType	เรียกค่าไทป์สำหรับอินสแตนซ์ปัจจุบัน
ReferenceEquals	ใช้ในการเช็คว่าออบเจกต์นั้น ๆ เป็นออบเจกต์ตัวเดียวกันหรือไม่
ToString	ส่งค่าสตริงที่เป็นค่าปัจจุบันของออบเจกต์นั้น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 แสดงเมทอดที่มีการเข้าถึงเป็น private

Name	Description
Finalize	เป็นคำสั่งที่อนุญาตให้ออบเจกต์ปล่อยพื้นที่ที่จองไว้ ก่อนที่จะถูกจัดการด้วยการเบจคอลเลคชัน
MemberwiseClone	สร้าง โครงก๊อปปี้สำหรับออบเจกต์นั้น ๆ

2.2 RDBMS

RDBMS หรือ Relational Database Management System ฐานข้อมูลชนิดนี้เป็นระบบจัดการฐานข้อมูลที่ใช้กันอยู่ในปัจจุบัน และมีการใช้งานที่หลากหลายที่สุด ตัว RDBMS นั้นถูกสร้างขึ้นบนฐานของโครงสร้างฐานข้อมูล ซึ่งก็คือกลุ่มของข้อมูลที่มีความสัมพันธ์กัน โดยจะถูกควบคุมด้วยกฎ 3 ประการคือ ข้อมูลทุกอย่างจะอยู่ในรูปของตาราง กฎข้อที่สองคือการที่ ข้อมูลที่อยู่บนตารางจะต้องไม่มีความซ้ำซ้อน และข้อที่ 3 คือการใช้ Sql

ข้อดีของการใช้ดาต้าเบสแบบนี้คือระบบที่ได้มีความง่าย ยืดหยุ่นและมีผลผลิตที่ดี เพราะตารางนั้นง่ายต่อการสร้างความสัมพันธ์กับข้อมูลอื่น ซึ่งการเรียนรู้ในการใช้ SQL ก็เป็นเรื่องง่ายและการใช้ RDBMS นี้ผู้ใช้ต้องการเปลี่ยนแปลงดาต้าจะไม่มีการบอกผลกระทบคือ โปรแกรมหรือแอปพลิเคชัน ซึ่งในงานนี้เราจะทำการเชื่อมต่อกับ DBMS ผ่าน .NET จึงจำเป็นที่จะต้องมีการศึกษาเพิ่มเติม

2.2.1. SqlConnection class ตัว class นี้เมื่อเป็น object จะแสดงถึงเซสชันในการเชื่อมต่อกับข้อมูลใน Sql ดาต้าเซิร์ฟเวอร์ โดยจะใช้ร่วมกันกับ class SqlDataAdapter และคลาส SqlCommand เพื่อที่จะเพิ่มประสิทธิภาพในการทำงาน ที่จะเชื่อมต่อและจัดการกับข้อมูลในเซิร์ฟเวอร์ Sql ของไมโครซอฟท์ ซึ่งมีเมมเบอร์ทั้งหมดของคลาสดังนี้

ตารางที่ 2.5 แสดง Public Constructors

Name	Description
SqlConnection	สร้างอินสแตนซ์ใหม่สำหรับคลาส sqlconnection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.6 แสดง Public Properties

Name	Description
<u>ConnectionString</u>	ใช้เรียกดูหรือสร้างสตริงที่เขียนข้อมูลของ คาด้าเบส เซิร์ฟเวอร์
<u>ConnectionTimeout</u>	ใช้เรียกดูหรือระบุช่วงเวลาที่ใช้ในการสร้างการเชื่อมต่อกับคาด้าเบสเซิร์ฟเวอร์ก่อนที่จะตัดการเชื่อมต่อ
Container	เรียกค่า IContainer ที่เก็บค่าของคอมเนนท์นั้น ๆ
<u>Database</u>	เรียกค่าชื่อของคาด้าเบสที่ทำการเชื่อมต่ออยู่ในขณะนั้น
<u>DataSource</u>	เรียกค่าชื่อของอินสแตนซ์ของ SQL เซิร์ฟเวอร์ที่ทำการเชื่อมต่อ
<u>FireInfoMessageEventOnUserErrors</u>	เรียกดูหรือสร้างคุณสมบัติ <u>FireInfoMessageEventOnUserErrors</u>
<u>PacketSize</u>	เรียกดูขนาดของแพคเกจที่ใช้ในการสื่อสารกับ อินสแตนซ์ของ SQL เซิร์ฟเวอร์
<u>ServerVersion</u>	เรียกค่า เวอร์ชันของอินสแตนซ์ของ SQL เซิร์ฟเวอร์ที่กำลังเชื่อมต่ออยู่
Site	เรียกค่าหรือตั้งค่า ISite
<u>State</u>	แสดงสถานะของ SqlConnection
<u>StatisticsEnabled</u>	ถ้าเซ็ทค่าให้เป็นจริงจะเป็นการเปิดการทำงานแบบสถิติเกเทอร์ริง(static gathering)
<u>WorkstationId</u>	เรียกค่าสตริงข้อมูลของคาด้าเบสไคลเอนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.7 แสดง Protected Properties

Name	Description
CanRaiseEvents	เรียกค่าที่ระบุให้ คอมโพเนนท์ใด ๆ สามารถเรียกอีเวนต์ออกมาได้
DesignMode	เรียกค่าสำหรับเช็คว่ คอมโพเนนท์นั้นอยู่ในโหมดดีไซน์หรือไม่
Events	เรียกค่าของอีเวนต์ทั้งหมดที่มีในคอมโพเนนท์นี้

ตารางที่ 2.8 แสดง Public Methods

Name	Description
<u>BeginTransaction</u>	เริ่มคำสั่งเบสทรานแซคชัน(transaction)
<u>ChangeDatabase</u>	เปลี่ยนคำสั่งเบสสำหรับการเชื่อมต่อปัจจุบัน
<u>ChangePassword</u>	เปลี่ยนพาสเวิร์ด ของ SQL เซิร์ฟเวอร์
<u>ClearAllPools</u>	ล้างค่าของ connection pool ทั้งหมด
<u>ClearPool</u>	ล้างค่าของ connection pool ที่ระบุไว้ใน connection
<u>Close</u>	ปิดการเชื่อมต่อกับคำสั่งเบสปัจจุบัน
<u>CreateCommand</u>	สร้าง SQL Command
CreateObjRef	สร้างออบเจกต์ที่บรรจุข้อมูลที่ใช้ในการสร้างพรอกซีสำหรับเชื่อมต่อกับออบเจกต์
Dispose	ปล่อยทรัพยากรที่ใช้โดยคอมโพเนนท์
<u>EnlistDistributedTransaction</u>	สร้างรายชื่อของ ทรานแซคชันที่กำหนดให้เป็น distributed ทรานแซคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.8 (ต่อ)

EnlistTransaction	สร้างรายชื่อของ ทรานแซกชันที่กำหนดให้เป็น distributed ทรานแซกชัน
Equals	ใช้ตรวจสอบค่าของออบเจกต์ทั้งสองมีค่าเท่ากันหรือไม่
GetHashCode	ใช้เรียก GetHashCode สำหรับการหาฟังก์ชัน hash .
GetType	เรียกค่าไทป์สำหรับอินสแตนซ์ปัจจุบัน
ReferenceEquals	ใช้ในการเช็คค่าของออบเจกต์นั้น ๆ เป็นออบเจกต์ตัวเดียวกันหรือไม่
ToString	ส่งค่าสตริงที่เป็นค่าปัจจุบันของออบเจกต์นั้น ๆ
GetLifetimeService	ใช้รับค่า lifetime ที่ใช้ใน lifetime control policy ของออบเจกต์ปัจจุบัน
Getschema	คืนค่าข้อมูล Schema สำหรับดาต้าเบสที่เชื่อมต่อ
InitializeLifetimeService	กำหนดค่าเริ่มต้นสำหรับ lifetime ที่ใช้ใน lifetime control policy ของออบเจกต์ปัจจุบัน
Open	ใช้ในเปิดการเชื่อมต่อกับ SQL เซิร์ฟเวอร์
ResetStatistics	ถ้าค่า statistic gathering ถูกเซ็ทไว้ก่อนหน้านี้จะถูกรีเซ็ทให้เป็นศูนย์ทั้งหมด
RetrieveStatistics	คืนค่าของชื่อการจับคู่คอลเลกชัน ในขณะที่เมทอดได้ถูกเรียกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.9 แสดง Protected Methods

Name	Description
Dispose	คืนค่าทรัพยากรทั้งหมดที่อินสแตนซ์นี้ใช้
Finalize	เป็นคำสั่งที่อนุญาตให้ออบเจกต์ปล่อยพื้นที่ที่จองไว้ก่อนที่จะถูกจัดการด้วยคาร์เบจคอลเลกชัน
GetService	คืนค่าออบเจกต์ที่แสดงเซอร์วิสของออบเจกต์นั้น ๆ
MemberwiseClone	สร้าง โครงก๊อปปี้สำหรับออบเจกต์นั้น ๆ

ตารางที่ 2.10 แสดง Public Events

Name	Description
Disposed	เพิ่มให้ตัวจัดการอีเวนต์คอยดูการเรียกใช้อีเวนต์ dispose
InfoMessage	จะมีปรากฏค่าเมื่อSQL เซิร์ฟเวอร์ส่งค่าค่าเตือนหรือข้อมูลกลับมา
StateChange	จะปรากฏค่าเมื่อสถานะของอีเวนต์มีการเปลี่ยนแปลง

ตารางที่ 2.11 แสดง Explicit Interface Implementations

Name	Description
<u>System.Data.IDbConnection.BeginTransaction</u>	
<u>System.Data.IDbConnection.CreateCommand</u>	
<u>System.ICloneable.Clone</u>	สร้างออบเจกต์ใหม่ที่เป็น โคลนของอินสแตนซ์เดิม
<u>System.IDisposable.Dispose</u>	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 Log File

ล็อกไฟล์ คือ ไฟล์ที่เก็บบันทึกกิจกรรมการดำเนินงานที่เกิดขึ้น โดย application จะเรียกข้อมูล “log” ว่า ล็อกไฟล์ในทางเทคนิคถ้ามีปัญหาอะไรเกิดขึ้นก็จะเปิดเรียกดู ล็อกไฟล์เพื่อหาสาเหตุของปัญหาและแก้ไข นอกจากนี้ข้อมูลใน ล็อกไฟล์ยังสามารถนำไปวิเคราะห์เป็นสถิติหรือรายงานเพื่อปรับปรุงประสิทธิภาพการทำงานให้ดีขึ้น

ในอินเทอร์เน็ตจะมี เซิร์ฟเวอร์ ที่ให้บริการมากมาย อย่างเช่น เว็บเซิร์ฟเวอร์ ก็จะมีการสร้าง ล็อกไฟล์โดยข้อมูลใน ล็อกไฟล์ก็จะเกี่ยวกับการร้องขอบริการที่มีเข้ามาใน เซิร์ฟเวอร์ เช่น เวลาที่มีการร้องขอบริการ, ใครที่ร้องขอบริการ, ไฟล์ไหนที่ถูกให้บริการ

ล็อกไฟล์โดยทั่วไปจะใช้นามสกุล .log แต่ก็ไม่จำเป็นเสมอไปว่าจะต้องใช้นามสกุลเป็น .log ขึ้นอยู่กับว่าใช้ application ไหน

2.4 Log File in Apache HTTP Server

Apache คือ open –source เว็บเซิร์ฟเวอร์ โดย Apache HTTP Server เป็นหนึ่งใน เซิร์ฟเวอร์ที่นิยมใช้ในอินเทอร์เน็ต

การจัดการ เว็บเซิร์ฟเวอร์ ให้มีประสิทธิภาพนั้นจำเป็นที่จะต้องได้รับรู้ผลการดำเนินงานและปัญหาต่างๆที่เกิดขึ้นใน เซิร์ฟเวอร์ ตัว Apache HTTP Server มีความยืดหยุ่นและครอบคลุมการ logging ได้อย่างมีประสิทธิภาพ

2.4.1 Error Log

error log จะถูกกำหนดชื่อและที่อยู่โดย ErrorLog directive ใน error log จะเป็นที่ตั้งซึ่ง Apache httpd ส่งข้อมูลที่พิจารณาและบันทึกข้อผิดพลาดต่างๆที่พบเจอ ในกระบวนการ request และจะเป็นที่แรกที่ดูปัญหาที่เกิดขึ้นเมื่อ เซิร์ฟเวอร์ เริ่มทำงาน โดยทั่วไปจะเก็บรายละเอียดของข้อผิดพลาด

โดยปกติ error log จะถูกเขียนลงไฟล์(error_log บน Unix systems และ error.log บน Windows และ OS/2)

รูปแบบของ error log ก่อนข้างที่จะเป็นแบบที่อิสระและมีการบรรยาย แต่ก็มีข้อมูลที่จำเป็นแน่นอนที่จะต้องถูกเก็บใน error log

```
[Wed Oct 11 14:32:52 2000] [error] [client 127.0.0.1] client
denied by server configuration: /export/home/live/ap/htdocs/test
```

รูปที่ 2.4 ตัวอย่าง error log

ส่วนแรกของตัวอย่างจะเป็นวันและเวลาของ message ส่วนที่สองเป็นรายการของข้อผิดพลาดที่เป็นรายงาน โดย LogLevel directive จะเป็นตัวที่ควบคุมชนิดของข้อผิดพลาดและส่งให้ error log ส่วนที่สามจะเป็น IP address ของ ไคลแอน ที่ทำให้เกิดข้อผิดพลาด และส่วนที่เหลือก็เป็นกรณีของข้อผิดพลาดและ พารามิเตอร์ของเอกสารที่ถูกขอ

Error log ยังมีรูปแบบที่หลากหลายและแตกต่างกันมากมายแต่ส่วนใหญ่ก็จะมีรูปแบบที่คล้ายกับตัวอย่างที่แสดง error log อาจจะยังคงต้องเก็บผลของการ debugging จาก CGI script ข้อมูลใดๆที่ถูกเขียนลง stderr โดย CGI script จะถูกก็อปไปใส่ error log โดยตรง

เป็นไปได้ที่จะปรับเปลี่ยน error log โดยการเพิ่มหรือลบข้อมูล แต่อย่างไรก็ตามสามารถจัดการ error log ที่มีเฉพาะ request ที่มีลักษณะเช่นเดียวกันใน access log

2.4.2 Access Log

access log จะบันทึกกระบวนการ request ทั้งหมดที่ส่งมายัง เซิร์ฟเวอร์ การกำหนดที่ตั้งและรายละเอียดของ access log จะถูกควบคุมโดย CustomLog directive ส่วน LogFormat directive ใช้ในการเลือกรายละเอียดของ logs

รูปแบบของ access log เป็นการกำหนดค่าขั้นสูง ซึ่งถูกระบุให้ใช้รูปแบบของสตริง ซึ่งรูปแบบที่ถูกกำหนดขึ้นใน access log จะมี 2 แบบ คือ Common Log Format และ Combined Log Format

Common Log File Format

```
LogFormat "%h %l %u %t \"%r\" %>s %b" common
161.246.243.152 - - [30/Sep/2007:05:26:17 +0700] "GET /index/index.php HTTP/1.1" 200 25030
```

รูปที่ 2.5 ตัวอย่าง Common Log File Format

แต่ละส่วนของ Common Log File Format มีรายละเอียดดังนี้

161.246.243.152 (%h)

IP Address ของ ไคลแอน ที่ร้องขอบริการจาก เซิร์ฟเวอร์

- (%l)

Remote logname โดยส่วนใหญ่มักจะไม่ระบุค่า("-")

- (%u)

userid ของบุคคลที่ร้องขอ document

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[30/Sep/2007:05:26:17 +0700] (%t)

เวลาที่เซิร์ฟเวอร์ได้รับ request มีรูปแบบดังนี้

[day/month/year:hour:minute:second zone]

day = 2*digit

second = 2*digit

month = 3*letter

zone = ('+' | '-') 4*digit

year = 4*digit

hour = 2*digit

minute = 2*digit second = 2*digit

"GET /index/index.php HTTP/1.1" ("%r\")

Request Line ที่ไคลแอนต์ส่งไปให้เซิร์ฟเวอร์ โดยมีส่วนประกอบดังนี้

Request Method : GET

Transport Protocol : HTTP/1.1

Request Stem : /index/index.php

Request Query String : ในตัวอย่างนี้ไม่มี

200 (%>s)

Status code ที่เซิร์ฟเวอร์ส่งกลับไปที่ไคลแอนต์

25030 (%b)

จำนวนไบต์ของ object ที่เซิร์ฟเวอร์ส่งกลับมาให้ไคลแอนต์

Combined Log Format

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-agent}i\"" combined
161.246.243.152 - - [30/Sep/2007:05:26:17 +0700] "GET /index/index.php HTTP/1.1" 200 25030
"http://www.kmitl.ac.th/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR
1.0.3705; .NET CLR 1.1.4322; Media Center PC 4.0)"
```

รูปที่ 2.6 ตัวอย่าง Combined Log Format

Combined Log Format มีรูปแบบเหมือน Common Log Format โดยจะเพิ่มเติมจาก

Common Log Format อีกสองส่วนคือ Referrer และ User Agent

"http://www.kmitl.ac.th/" (^%{Referer}i")

เว็บเพจที่ให้ลิงค์เชื่อมต่อมาที่เว็บไซต์

"Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.0.3705;

.NET CLR 1.1.4322; Media Center PC 4.0)" (^%{User-agent}i")

ข้อมูลเกี่ยวกับ browser ที่ขอ request โดยมีส่วนประกอบดังนี้

Browser : MSIE 6.0 คือ Internet Explorer 6.0

Browser Operating System : Windows NT 5.1 คือ Windows XP

2.4.3 Multiple Access Logs

Multiple access log สามารถสร้างได้ง่ายๆโดยการกำหนดค่าของ CustomLog directive หลายๆค่า ใน configuration ไฟล์

```
LogFormat "%h %l %u %t \"%r\" %>s %b" common
CustomLog logs/access_log common
CustomLog logs/referer_log "%{Referer}i -> %U"
CustomLog logs/agent_log "%{User-agent}i"
```

รูปที่ 2.7 ตัวอย่างการสร้าง Multiple Access Log

จากตัวอย่างจะเห็นว่ามีการสร้างสาม access log โดย access log แรกจะเป็นการเก็บข้อมูลแบบ Common Log Format ในขณะที่ access log ที่สองจะเก็บข้อมูลของ referer และ access log ที่สามจะเก็บข้อมูล browser

2.4.4 Log Rotation

ในเวลาที่ เซิร์ฟเวอร์ ทำงานหนักจะมีปริมาณข้อมูลที่ถูกเก็บใน ล็อกไฟล์เป็นจำนวนมาก โดยทั่วไป access log ไฟล์จะเพิ่มขึ้น 1 MB ต่อ 10,000 request เพราะฉะนั้นจึงมีความจำเป็นที่จะต้อง rotate ล็อกไฟล์เป็นบางครั้งบางคราว โดยการย้ายหรือลบล็อกไฟล์ที่มีอยู่ การ rotate ไม่สามารถทำได้ในขณะที่ เซิร์ฟเวอร์ กำลังทำงานอยู่เนื่องจาก Apache จะเขียนลงล็อกไฟล์เก่าต่อไปเรื่อยๆในขณะที่ไฟล์เปิดอยู่ ดังนั้นถ้าจะทำการ rotate ได้ นั้น เซิร์ฟเวอร์ จะต้องถูก restart หลังจากนั้นล็อกไฟล์จะถูกย้ายหรือลบแล้ว จึงค่อยเปิดล็อกไฟล์ใหม่

ในการ restart นั้นสามารถสั่งให้ เซิร์ฟเวอร์ เปิดล็อกไฟล์ใหม่โดยไม่สูญเสียการเชื่อมต่อกับ ไคลเอน ที่มีอยู่หรือที่คอยอยู่ แต่อย่างไรก็ตามจะสามารถทำได้เมื่อ เซิร์ฟเวอร์ ต้องเขียนล็อกไฟล์เก่าต่อไปในขณะที่ให้บริการ request เก่าไปแล้ว ต่อจากนั้นก็จำเป็นที่จะต้องรอหลังจากทำการ restart ก่อนที่จะทำการระบวนการต่างๆกับล็อกไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.5 Piped Log

ใน Apache มีความสามารถในการเขียน error log และ access log ผ่าน pipe ไปยังโปรเซสอื่น การทำอย่างนี้จะเป็นการเพิ่มความคล่องตัวของ logging โดยที่ไม่ต้องเพิ่มโค้ดลงไป เซิร์ฟเวอร์หลัก คำสั่งที่ใช้ให้เขียนล็อกไปยัง pipe ทำได้ง่ายๆ โดยการแทนที่ชื่อไฟล์ด้วยตัวอักษร “|”(pipe)ตามด้วยชื่อไฟล์ที่สามารถ execute ได้ Apache เริ่มกระบวนการ pipe-log เมื่อ เซิร์ฟเวอร์ เริ่มต้นทำงานและจะเริ่มทำกระบวนการใหม่ถ้าเกิดการชนกันในขณะที่ เซิร์ฟเวอร์ ทำงาน

2.5 Log File in Internet Information Services(IIS)

Internet Information Services(IIS) ได้เสนอทางเลือกสำหรับบันทึกกิจกรรมที่เกิดขึ้นใน Web sites, File Transfer Protocol (FTP) sites, Network News Transfer Protocol (NNTP) service และ Simple Mail Transfer Protocol (SMTP) service และยังอนุญาตให้เลือกรูปแบบของล็อกไฟล์ที่คิดว่าเหมาะสมที่สุดกับสถานการณ์ IIS ล็อกไฟล์ประกอบด้วยข้อมูลต่างๆ เช่น ใครเข้ามาที่เว็บไซต์ เข้ามาเมื่อไร จึงทำให้เราสามารถตรวจสอบว่าบริการนี้มีความต้องการมากแค่ไหน โดย IIS จะบันทึกเหตุการณ์ต่างที่เกิดขึ้นในการให้บริการ

2.5.1 W3C Extended Log Format

Log format นี้ถูกใช้สำหรับ Microsoft IIS ภายในประกอบด้วยบรรทัดของตัวอักษร ASCII ต่อเนื่องกัน โดยแต่ละบรรทัดนั้นจะมี directive หรือ entry

Entries ประกอบขึ้นมาจากหลาย ๆ fields ซึ่งแต่ละ fields จะมีข้อมูลจาก HTTP TRANSACTION ซึ่งจะแยกออกจากกันด้วยเครื่องหมายวรรคตอน และถ้า fields นั้นไม่มีการใช้งานก็ถูกแทนที่ด้วยเครื่องหมาย - ใน fields นั้น ๆ

Directives บันทึกข้อมูลของการบันทึก ล็อกไฟล์โดยแต่ละบรรทัดนั้นจะมีเครื่องหมาย # นำหน้าเพื่อเป็นสัญลักษณ์ในการแยกออกจาก Entries โดยลักษณะการเก็บข้อมูลของ directives ของ W3C Extended format มีรายละเอียดดังนี้

Version: <integer>.<integer>	Version ของ extended log file format ที่ใช้
Fields: [<specifier>...]	กำหนด field ที่จะบันทึกลงในล็อกไฟล์
Software: string	
Start-Date: <date> <time>	วันเวลาที่ล็อกไฟล์ เริ่มทำงาน
End-Date: <date> <time>	วันเวลาที่ล็อกไฟล์ ทำงานเสร็จ
Date: <date> <time>	วันเวลาที่ entry ถูกบันทึก
Remark: <text>	ข้อคิดเห็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Version และ Fields directives ควรจะนำหน้า entry ทั้งหมดในล็อกไฟล์ โดย Field directive จะกำหนดข้อมูลที่จะบันทึกลงใน field ของแต่ละ entry

```
#Software: Microsoft Internet Information Services 6.0
#Version: 1.0
#Date: 2002-05-24 20:18:01
#Fields: date time c-ip cs-username s-ip s-port cs-method cs-uri-stem cs-uri-query sc-status sc-bytes
cs-bytes time-taken cs(User-Agent) cs(Referrer)
2002-05-24 20:18:01 172.224.24.114 - 206.73.118.24 80 GET /Default.htm - 200 7930 248 31
Mozilla/4.0+(compatible;+MSIE+5.01;+Windows+2000+Server) http://64.224.24.114/
```

รูปที่ 2.8 ตัวอย่าง W3C Extend Log File จาก Internet

จากตัวอย่าง W3C Extend Log File จาก Internet ในส่วนของ Field สามารถอธิบายได้ดังนี้

- **date** วันเดือนปีที่ ล็อกไฟล์ถูกบันทึก
- **time** เวลาที่ ล็อกไฟล์ถูกบันทึก
- **c-ip** IP address ของ ไคลเอนต์
- **cs-username** user name ที่เข้าใช้ เซิร์ฟเวอร์
- **s-ip** IP address ของ เซิร์ฟเวอร์
- **s-port** เซิร์ฟเวอร์ port
- **cs-method** HTTP method ที่ใช้ในการ request
- **cs-uri-stem** Request Stem
- **cs-uri-query** Request Query string
- **sc-status** status code ที่ เซิร์ฟเวอร์ ส่งกลับไปให้ ไคลเอนต์
- **sc-bytes** จำนวนไบต์ที่ เซิร์ฟเวอร์ ส่งไปให้ ไคลเอนต์
- **cs-bytes** จำนวนไบต์ที่ ไคลเอนต์ ส่งไปให้ เซิร์ฟเวอร์
- **time-taken** เวลาที่ใช้ในการ request
- **cs(User-Agent)** ข้อมูลเกี่ยวกับ browser ที่ขอ request
- **cs(Referrer)** เว็บเพจที่ให้ลิงค์เชื่อมต่อมาที่เว็บไซต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 IIS Log File Format

รูปแบบของ IIS log file format จะเป็นแบบ fixed ASCII text-based format โดยที่ไม่สามารถปรับเปลี่ยนไปตามความต้องการของผู้ใช้ เนื่องจาก IIS log file format ถูกควบคุมโดย HTTP.sys

```
192.168.114.201, -, 03/20/01, 7:55:20, W3SVC2, SERVER, 172.21.13.45, 4502, 163, 3223, 200, 0,
GET, /DeptLogo.gif, -,
```

รูปที่ 2.9 ตัวอย่าง IIS ล็อกไฟล์

จากตัวอย่าง IIS ล็อกไฟล์แต่ละ field มีรายละเอียดดังนี้

- **Client IP address** : 192.168.114.201 (IP address ของ ไคลแอน)
- **User name** : -
- **Date** : 03/20/01
- **Time** : 7:55:20
- **Service and instance** : W3SVC2
- **Server name** : SERVER
- **Server IP** : 172.21.13.45 (IP address ของ เซิร์ฟเวอร์)
- **Time taken** : 4502 (หน่วยเป็น milliseconds)
- **Client bytes sent** : 163 (จำนวนไบต์ที่ส่งจาก ไคลแอน ไป เซิร์ฟเวอร์)
- **Server bytes sent** : 3223 (จำนวนไบต์ที่ส่งจาก เซิร์ฟเวอร์ ไป ไคลแอน)
- **Service status code** : 200 (การร้องขอสมบูรณ์เรียบร้อย)
- **Windows status code** : 0 (การร้องขอสมบูรณ์เรียบร้อย)
- **Request type** : GET
- **Target of operation** : /DeptLogo.gif (user ต้องการ download ไฟล์ DeptLogo.gif)
- **Parameters** : -

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 Log File in Sun™ ONE Web Server(iPlanet)

ล็อกไฟล์ใน One Web Server (iPlanet) format ประกอบด้วยบรรทัดของตัวอักษร ASCII ต่อเนื่องกัน โดยแต่ละบรรทัดจะมี directive หรือ entry

Entries ประกอบขึ้นมาจากหลาย ๆ fields ซึ่งแต่ละ fields จะมีความสัมพันธ์กับ HTTP TRANSACTION ซึ่งจะแยกออกจากกันด้วยเครื่องหมายวรรคตอน และถ้า fields นั้นไม่มีการใช้งานก็ถูกแทนที่ด้วยเครื่องหมาย - ใน fields นั้น ๆ

Directive จะปรากฏอยู่ในส่วนต้นของ ล็อกไฟล์โดยจะกำหนดข้อมูลและรูปแบบที่จะเก็บใน entry

```
format=%Ses->client.ip% - %Req->vars.auth-user% [%SYSDATE%]
"%Req->reqpb.clf-request%" %Req->srvhdrs.clf-status%
%Req->srvhdrs.content-length%
10.101.128.66 - - [03/Apr/2002:00:42:11 -0500] "GET /multimedia/flash/dis.swf HTTP/1.1" 304
```

รูปที่ 2.10 ตัวอย่าง Sun™ ONE Web Server (iPlanet) log format

2.7 WebSphere Application Server (WAS) Log Format

WebSphere Application Server (WAS) version 4.0 ได้จัดเตรียม API สำหรับ application level logging เรียกว่า Analytic Logging Service (ALS) โดยเราสามารถเรียกใช้ API นี้จาก Web application ของเราเอง IBM WebSphere Personalization server ใช้ API นี้กับข้อมูลล็อกไฟล์

API นี้สนับสนุน logging method 3 อย่างด้วยกันคือ HTTP , database และ file รูปแบบการเก็บข้อมูลของ ล็อกไฟล์จะแตกต่างจาก logging method ที่ใช้เพียงเล็กน้อย อย่างไรก็ตามการเก็บบันทึกล็อกไฟล์จะต้องมี 5 field คือ appid , reqid , httpdata , cookie และ appdata

```
<custCareID> <01010123-1234> <-> <-> <target=TP600E;crosssells=TP770,TP380;ads=pIII>
```

รูปที่ 2.11 ตัวอย่างการเก็บบันทึกล็อกไฟล์ใน field ทั้ง 5 ของ WAS

จากตัวอย่างอธิบายรายละเอียดของแต่ละ field ได้ดังนี้

appid <custCareID> - Application ID ที่ทำการบันทึก

reqid <01010123-1234> - HTTP request ที่เกี่ยวข้องกับ ID โดย reqid ถูกใช้สำหรับ ความสัมพันธ์ของ JSP request ที่ให้บริการ HTTP request

httpdata <-> - Combined log

cookie <-> - ข้อมูลของ cookie

appdata <target=TP600E;crosssells=TP770,TP380;ads=pIII> - Appdata(application data) field นี้ เก็บข้อมูลของ application level โดยข้อมูลใน field นี้ จะเป็น list ที่มีรูปแบบเป็น key = values ต่อเนื่องกัน โดยมี semicolon แบ่งคั่นเป็นชุด ๆ โดย key นั้นจะแสดงค่าของ semantic label ส่วน values นั้นก็จะแสดงค่า semantic v alues นั้น ๆ โดยคั่นด้วยเครื่องหมายคอมม่า และถ้า ใน values มีเครื่องหมาย comma ประกอบอยู่ comma นั้นจะต้องมีเครื่องหมาย “ ” ครอบมอยู่

2.8 ผลการศึกษาโปรแกรมวิเคราะห์ล็อกไฟล์ที่มีอยู่ในปัจจุบัน

โปรแกรมที่สามารถทำการวิเคราะห์ล็อกไฟล์ในปัจจุบันนั้น ได้มีผู้ทำการพัฒนาขึ้นมา มาก - มาก ส่วนใหญ่แล้วก็จะจะมี Features หลัก ๆ ที่คล้ายคลึงกันเช่น ต้องสามารถดู จำนวนผู้เข้าใช้ จำนวน ผู้เข้าใช้ในวันเวลาต่าง ๆ เบราเซอร์ที่ใช้ ฯลฯ จึงได้สรุปความสามารถของโปรแกรมที่นับว่าได้รับความนิยมนสูง และเป็น open source เพื่อนำมาเป็นแนวทางในการพัฒนาโปรแกรมต่อไป โดยข้อมูลที่ศึกษามาสามารถสรุปได้ดังนี้

ตารางที่ 2.12 แสดงรายละเอียดของตาราง ข้อมูลของโปรแกรม log analyzer ในปัจจุบัน

Feature/software	Awstats	Analog	Webalizer
version	6.5	6.0	2.01
ภาษาที่ใช้พัฒนา	Perl	C	C
ระบบปฏิบัติการที่รับรอง	ทุกระบบ	ทุกระบบ	ทุกระบบ
ราคาและลิขสิทธิ์	Free/GPL	Free/GPL	Free/GPL
การทำงานกับ Apache combined log format	ทำงานได้	ทำงานได้	ทำงานได้
การทำงานกับ Apache common log format	ทำงานได้บางความสามารถโปรแกรม	ทำงานได้บางความสามารถโปรแกรม	ทำงานได้บางความสามารถโปรแกรม
การทำงานกับ IIS log format	ทำงานได้	ทำงานได้	ต้องใช้แพทช์เสริม
วิเคราะห์ mail/FTP log files	ทำได้ / ทำได้	ทำไม่ได้	ทำไม่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.12 (ต่อ)

ทำการอพยพข้อมูลผ่านทาง	คอมมานด์ ไลน์หรือเบรา เซอร์	คอมมานด์ ไลน์หรือเบรา เซอร์	คอมมานด์ ไลน์
รายงานจำนวนคนที่เข้าใช้	ทำได้	ทำได้	ทำไม่ได้
รายงานจำนวนคนเข้าใช้ที่ไม่ซ้ำ	ทำได้	ทำไม่ได้	ทำไม่ได้
รายงานระยะเวลาที่เข้าใช้งาน	ทำได้	ทำไม่ได้	ทำไม่ได้
รายงานการเข้าใช้แบ่งตามประเทศ	ทำได้	ทำได้	ทำได้
รายงานการเข้าใช้ตาม host	ทำได้	ทำได้	ทำได้
รายงานการเข้าใช้ของ robot หรือ spiders	ทำได้	ทำได้	ทำไม่ได้
รายงานช่วงเวลาที่เป็นชั่วโมงเร่งด่วน	ทำได้	ทำได้	ทำได้
รายงานการเข้าใช้ตามจำนวนวัน	ทำได้	ทำได้	ทำได้
รายงานเพจที่มีผู้เข้าใช้บ่อย	ทำได้	ทำได้	ทำได้
รายงานเพจที่เข้าสู่ไซต์	ทำได้	ทำไม่ได้	ทำได้
รายงานเพจด้วย เวลาที่เข้าใช้ล่าสุด	ทำได้	ทำได้	ทำไม่ได้
สามารถใส่ตัวคัดกรองที่ปรับแต่งได้สำหรับการวิเคราะห์ได้	ทำได้	ทำไม่ได้	ทำไม่ได้
รายงานระบบปฏิบัติการที่เข้าใช้	ทำได้	ทำได้	ทำไม่ได้
รายงานเบราเซอร์ที่เข้าใช้	ทำได้	ทำได้	ทำได้
รายงาน search engines ที่ใช้ในการค้นหาเว็บไซต์	ทำได้	ทำได้	ทำไม่ได้
รายงานคีย์เวิร์ดที่ใช้ในการค้นหา	ทำได้	ทำได้	ทำได้
รายงาน HTTP Errors	ทำได้	ทำได้	ทำได้
รายงานการ “add to favorites”	ทำได้	ทำไม่ได้	ทำไม่ได้
สรุปรายงานการเข้าใช้เป็นรายวัน	ทำได้	ทำได้	ทำได้
สรุปรายงานการเข้าใช้เป็นรายเดือน	ทำได้	ทำได้	ทำได้
สรุปรายงานการเข้าใช้เป็นรายปี	ทำได้	ทำได้	ทำได้
รูปแบบของรายงาน	Structured text file หรือ XML	Text file	Text file

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากข้อมูลข้างต้นจะเห็นได้ว่าแต่ละโปรแกรมนั้นก็มีข้อดีข้อเสียแตกต่างกันไปและมี features ที่น่าสนใจมากมาย แต่ว่าทั้งหมดนั้นมุ่งเน้นไปที่การวิเคราะห์เฉพาะเว็บไซต์หนึ่งเท่านั้นไม่สามารถวิเคราะห์ในลักษณะของ shared เซิร์ฟเวอร์ ได้ ซึ่งโปรแกรมที่จะพัฒนาขึ้นจึงมุ่งหมายให้สามารถทำงานในส่วนนี้ได้ด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบโปรแกรมและพัฒนา

ในการออกแบบและพัฒนานั้นจะใช้ทฤษฎีที่ศึกษาในบทที่ 2 มาใช้ในการออกแบบระบบ และพัฒนาระบบ ซึ่งจะเริ่มต้นจากการกำหนดแนวคิดในการออกแบบ กำหนดรายละเอียด ความสามารถของโปรแกรมให้ชัดเจน จากนั้นก็เป็นการออกแบบภาพรวมของโปรแกรม วิธีที่จะสร้างในแต่ละส่วนผ่านขั้นตอนของ UML จากนั้นก็เริ่มทำการโค้ดดิ้งและกำหนดรายละเอียดในส่วนต่าง ๆ ของโปรแกรม

3.1 แนวคิดในการออกแบบ

- เป็นโปรแกรมที่เข้าใ้การใช้งานได้ง่าย ไม่จำเป็นต้องมีความรู้พื้นฐานทางคอมพิวเตอร์มาก่อน ก็สามารถใช้งานได้ดี

- เป็นโปรแกรมที่ทำงานแบบ consolidate กล่าวคือ โปรแกรมสามารถรวม ล็อกไฟล์จากหลาย ๆ เซิร์ฟเวอร์ ทั้งประเภทเดียวกันหรือ ต่างประเภทกัน แล้วนำมาวิเคราะห์รวมกัน เนื่องจากในบางกรณีที่เซิร์ฟเวอร์ของเว็บที่ทำการวิเคราะห์นั้น บางครั้งจะทำงานในลักษณะของเซิร์ฟเวอร์ฟาร์ม เช่นในเซิร์ฟเวอร์ ของสำนักทะเบียน จึงจำเป็นที่ โปรแกรมจะต้องมีความสามารถนี้

- เป็นโปรแกรมที่เก็บข้อมูลทำงานในลักษณะของ database เพื่อที่จะได้นำส่วนดีของระบบจัดการคดีแบบสมาชิได้ ทำให้มีความยุ่งยากน้อยลง เมื่อจัดเก็บ ไปครั้งหนึ่งแล้วจะทำงานได้เร็วกว่า และมีความยืดหยุ่นสูงกว่าในการเรียกดูข้อมูลสำหรับผู้ที่มีพื้นฐานทางคอมพิวเตอร์

3.2 ความสามารถของโปรแกรม

- โปรแกรมสามารถรองรับ Apache common log format, Apache combined log format, IIS log format ได้

- โปรแกรมสามารถทำงานในการดึง ล็อกไฟล์มาจากหลาย ๆ เซิร์ฟเวอร์ ที่ทำงานร่วมกัน ในลักษณะ shared เซิร์ฟเวอร์ ได้

- โปรแกรมสามารถปรับแต่งส่วนที่ใช้ในการรับข้อมูล เพื่อกำหนดลักษณะที่เหมาะสมกับการใช้งานได้

- การส่งงานส่วนใหญ่ผ่าน GUI ที่สามารถเข้าใจได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-วิเคราะห์ข้อมูลจาก ล็อกไฟล์โดยแสดงตามเงื่อนไขต่าง ๆ ได้ตามรายการดังนี้

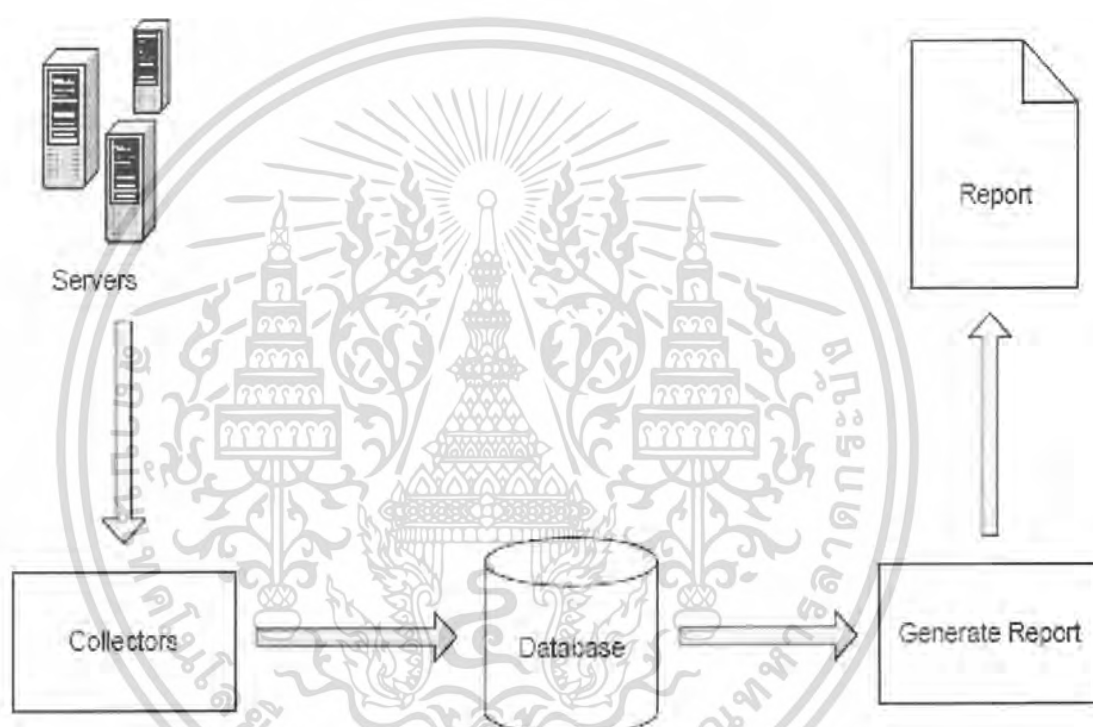
- รายงานจำนวนคนที่เข้าใช้
- รายงานจำนวนคนเข้าใช้ที่ไม่ซ้ำ
- รายงานระยะเวลาที่เข้าใช้งาน
- รายงานการเข้าใช้ตาม host
- รายงานการเข้าใช้ของ robot หรือ spiders
- รายงานการเข้าใช้ตามจำนวนวัน
- รายงานเพจด้วย เวลาที่เข้าใช้ล่าสุด
- สามารถใส่ตัวคัดกรองที่ปรับแต่งได้สำหรับการวิเคราะห์ได้
- รายงานระบบปฏิบัติการที่เข้าใช้
- รายงานเบราว์เซอร์ที่เข้าใช้
- รายงาน Search engines ที่ใช้ในการค้นหาเว็บไซต์
- รายงานคีย์เวิร์ดที่ใช้ในการค้นหา
- สรุปรายงานการเข้าใช้เป็นรายวัน
- สรุปรายงานการเข้าใช้เป็นรายเดือน
- สรุปรายงานการเข้าใช้เป็นรายปี

-สามารถแสดงข้อมูลที่วิเคราะห์นั้นด้วยสื่อที่สามารถเข้าใจและเปรียบเทียบได้ง่าย เช่น ในลักษณะของกราฟ

-สามารถบันทึกข้อมูลของรายงานที่ได้ในลักษณะของรายงานเพื่อใช้ในการใช้งานอื่น

3.3 ภาพรวมของโปรแกรม

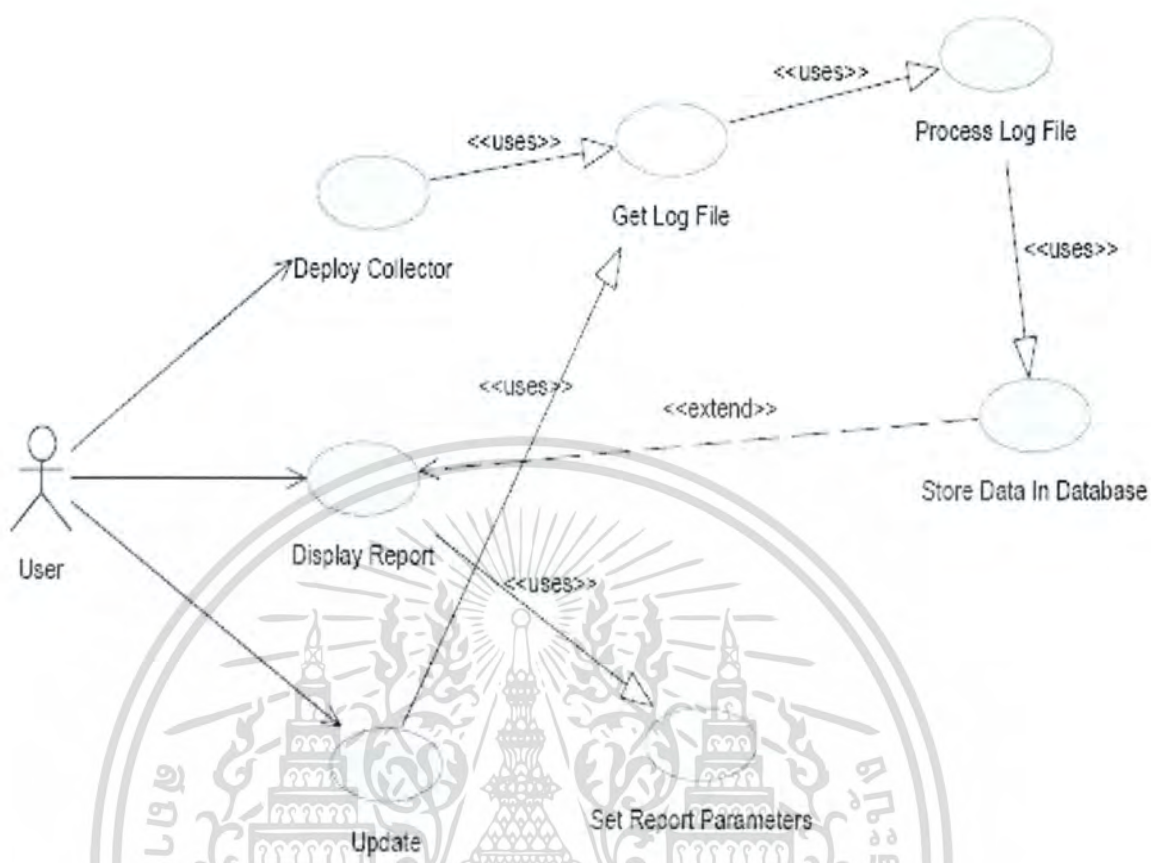
โปรแกรมจะประกอบไปด้วยสามส่วนหลักคือ Collector, database และ report ซึ่งจะทำงานโดย collector นั้นจะทำการดึง ล็อกไฟล์จาก เซิร์ฟเวอร์ ที่เราต้องการจากนั้นทำการคำนวณ และนำผลลัพธ์ที่ได้ซึ่งจะเป็น format มาตรฐานเดียวกันสำหรับทุก ๆ log format ไปเก็บไว้ใน database ต่อจากนั้นถ้าผู้ใช้ต้องการจะดูรายงานของ ล็อกไฟล์ที่ได้ผ่านการคำนวณแล้ว ส่วน Report ก็จะมีการสร้างรีพอร์ตขึ้นมาโดยนำข้อมูลจาก database ไปใช้ในการคำนวณ



รูปที่ 3.1แสดงโครงสร้างโดยรวมของโปรแกรม

ซึ่งในการออกแบบโปรแกรมนั้นจะอธิบายในรูปแบบของ UML ซึ่งโปรแกรมจะมียูสเคสไดอะแกรมดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดง Use Case Diagram

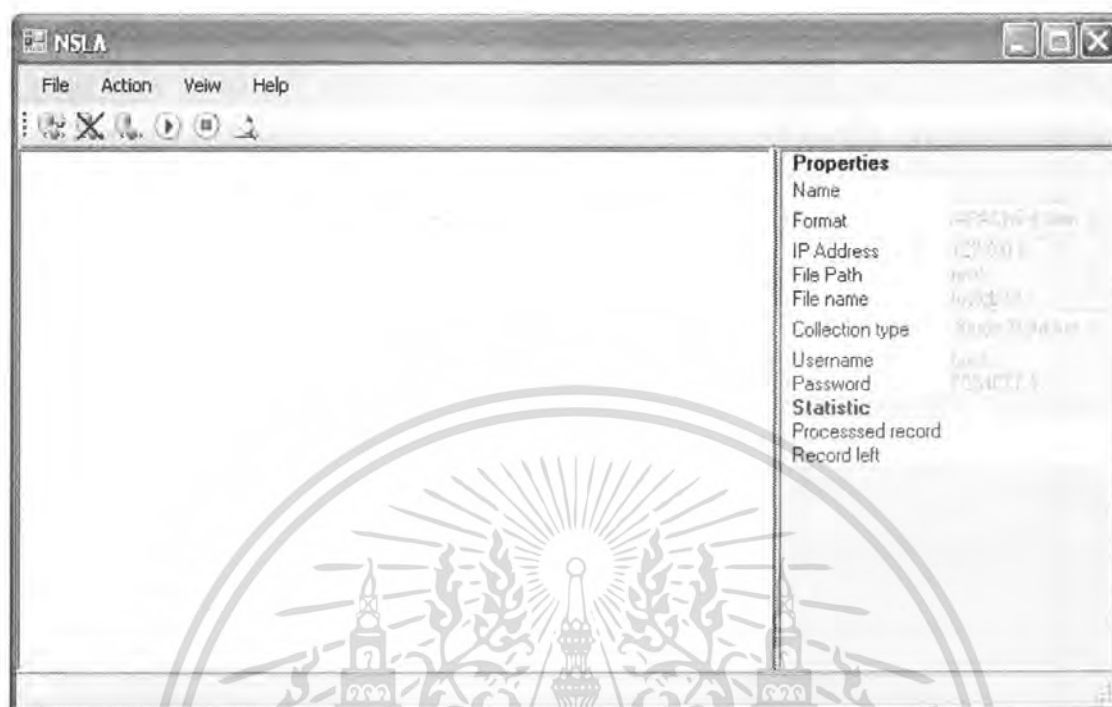
รายละเอียดต่าง ๆ ของยูสเคสไดอะแกรมอธิบายในตารางที่ 3.1

ตารางที่ 3.1 แสดงรายละเอียดของตารางยูสเคสไดอะแกรมของโปรแกรม

ยูสเคส	คำอธิบาย
Deploy collector	สร้าง, เชื่อม โยงและกำหนดค่าให้กับ collector
Display report	ดูรายงานที่ทำการคำนวณเรียบร้อยแล้ว
Update	ปรับข้อมูลของ ล็อกไฟล์ให้เป็นข้อมูลล่าสุด
Get log files	ดึง log files มาจาก sever ที่ระบุใน parameters ของ collector
Process log files	คำนวณ ล็อกไฟล์เพื่อให้เป็น standard format
Store data in database	เก็บ standard format ที่ได้ไว้ใน database
Set report parameters	กำหนดค่าเริ่มต้นของรายงาน เช่นช่วงเวลา ฯลฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

User Interface



รูปที่ 3.4 แสดงหน้าจอ interface ของโปรแกรม

หน้าจอของ โปรแกรมจะประกอบไปด้วย 4 ส่วนหลักคือ menu bar, tool bar, map panel และ properties panel

1.Menu bar ประกอบไปด้วยเมนูย่อยต่าง ๆ ดังนี้

1.1 File เป็นเมนูที่ใช้สำหรับการสร้าง เปิด หรือบันทึก โปรเจกต์ที่เราสร้างขึ้น

1.2 Action เป็นเมนูที่สั่งการเกี่ยวกับ collector โดยจะมีเมนูสำหรับการสร้าง ลบ และแก้ไข collector

1.3 View เป็นคำสั่งสำหรับดู ออก ไฟล์ที่กำลังทำการ process หรือค่าสถิติของ collector และดูรายงานของ เซิร์ฟเวอร์ ที่ประมวลผลแล้วได้

1.4 Help เป็นเมนูสำหรับเปิดตัวช่วยในการใช้งาน โปรแกรม

2.Tool bar มี tool button ทั้งหมด 5 ปุ่มดังนี้

2.1 Add collector เป็นคำสั่งสำหรับใช้สร้าง collector ตัวใหม่

2.2 Delete collector เป็นคำสั่งสำหรับใช้ในการลบ collector

2.3 Edit collector เป็นคำสั่งสำหรับใช้ในการแก้ไขค่าของ collector

2.4 Start เป็นคำสั่งที่สั่งให้ collector เริ่มการทำงาน

2.5 Stop เป็นคำสั่งที่สั่งให้ collector หยุดการทำงาน

2.6 View report เป็นคำสั่งที่เพื่อดู report

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Map panel เป็นหน้าจอที่จะแสดงถึงการเชื่อมโยงของ collector แต่ละตัว

4. Properties panel เป็นส่วนของหน้าจอที่ใช้แสดงและปรับพารามิเตอร์ทั้งหมดของ collector และแสดงสถานะการทำงานของ collector ในขณะนั้นด้วย

4.1. Properties Name เป็นส่วนที่กำหนดชื่อสำหรับ collector แต่ละตัวและจะเป็นชื่อเดียวกับ database ที่จะสร้างขึ้นด้วย โดยชื่อเหล่านี้จะไม่สามารถตั้งซ้ำกันได้

4.2. Properties Format จะเป็น combo box ที่แต่ละ item จะเก็บค่า format ของ ล็อกไฟล์แต่ละแบบไว้ ทั้งหมด 3 แบบคือ Apache – common, Apache – combined, IIS

4.3. Properties ID Address มีไว้สำหรับเก็บค่า IP Address ของ FTP Server ที่ทำการเก็บล็อกไฟล์ที่เราต้องการนำมาวิเคราะห์เอาไว้

4.4. Properties File Path มีไว้สำหรับเก็บค่า Location ของไฟล์ใน FTP Server ที่ทำการเก็บล็อกไฟล์ที่เราต้องการนำมาวิเคราะห์เอาไว้

4.5. Properties File Name มีไว้สำหรับเก็บค่าชื่อไฟล์ของ ล็อกไฟล์ที่เราต้องการนำมาวิเคราะห์

4.6. Properties collection type จะเป็น combo box ที่แต่ละ item จะเก็บค่าชนิดของ Collector ว่าจะทำงานแบบใดคือ หนึ่ง Single data source คือเป็นการดึงไฟล์มาวิเคราะห์จากเซิร์ฟเวอร์ตัวเดียว ส่วนแบบที่สองคือ Multiple data source เป็นการดึงไฟล์มาวิเคราะห์จากเซิร์ฟเวอร์แต่ละตัวและทำการวิเคราะห์ร่วมกันอีกครั้ง โดยอาศัย Collector แบบที่ 3 ที่เรียกว่า Collerator

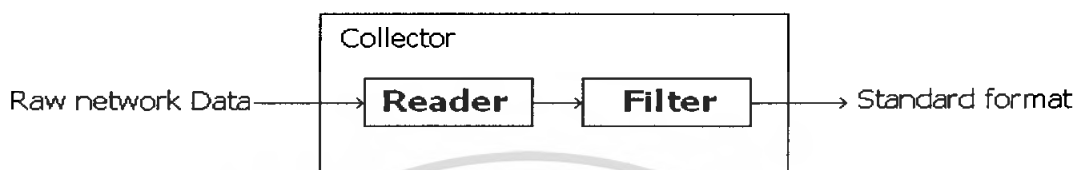
4.7. Properties User Name มีไว้สำหรับเก็บค่า User Name สำหรับเซิร์ฟเวอร์ที่เราดึง ล็อกไฟล์ที่ต้องการนำมาวิเคราะห์

4.8. Password มีไว้สำหรับเก็บค่า password สำหรับเซิร์ฟเวอร์ที่เราดึง ล็อกไฟล์ที่ต้องการนำมาวิเคราะห์

3.4 รายละเอียดแต่ละส่วนของโปรแกรม

3.4.1 Collector

Collector เป็น component หลักของโปรแกรมทำหน้าที่ในการรวบรวมและจัดการ ล็อกไฟล์ของ data source คัดกรองเฉพาะข้อมูลที่ต้องการและส่งต่อข้อมูลนั้นเพื่อนำไปประมวลผลต่อไป โดยมีองค์ประกอบหลัก 2 ส่วน คือ Reader และ Filter



รูปที่ 3.5 แสดงโครงสร้างของ collector

Reader ทำหน้าที่ในการติดต่อกับ data source ของระบบที่เราต้องทำการวิเคราะห์ เพื่อดึงข้อมูล ล็อกไฟล์แล้วนำมาเก็บไว้รอการ process โดยผู้ใช้งานจะต้องระบุข้อมูลของอุปกรณ์นั้น ๆ เช่น log format, ip address โดยส่วน reader ทำหน้าที่ดังกล่าวได้อย่างสมบูรณ์นั้นจะต้องทำงานได้มีลักษณะเหมือนเป็น FTP client เพื่อทำการติดต่อกับ FTP เซิร์ฟเวอร์ โดย Reader นั้นจะต้องมีความสามารถ ที่จะส่งสัญญาณเชื่อมต่อกับ FTP เซิร์ฟเวอร์ ใช้คำสั่งพื้นฐานต่าง ๆ ของ FTP เซิร์ฟเวอร์ได้ เช่น change directory, list ,cat และที่สำคัญที่สุดคือ get ซึ่งเราจะใช้ class FtpWebRequest ของ .NET ที่ได้กล่าวไว้แล้วเบื้องต้นในบทที่ 2 มาช่วยให้ส่วนของ reader นี้ทำงานได้ตามต้องการ ซึ่ง reader นั้นก็จะนำไฟล์ที่ดึงมาได้นั้นมาเก็บไว้ในเครื่อง แล้วให้ Filter ทำงานในส่วนของการตัดคำต่อไป

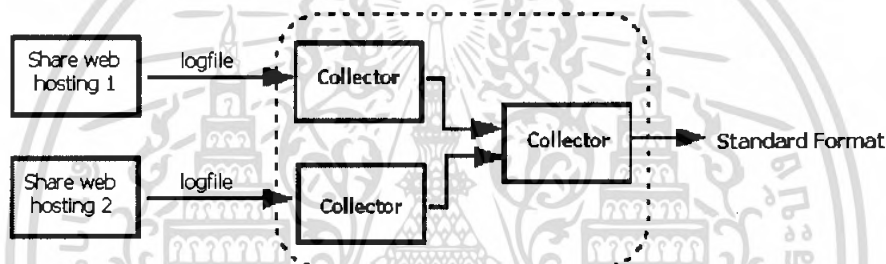
Filter เป็นส่วนที่ทำหน้าที่ประมวลผลข้อมูลที่ได้รับมาจาก reader โดยมีการทำงานหลากหลายรูปแบบดังนี้

- ทำการลดจำนวนข้อมูลที่มีการซ้ำซ้อนกันเช่น การรวบรวมข้อมูลที่มีหมายเลข ip เดียวกันเข้าด้วยกัน
- เพิ่มหรือลด fields ของแต่ละ record
- แบ่งแยกข้อมูลออกตามฟิลด์ของ standard format
- รวบรวมข้อมูลจาก collector ตัวอื่น ๆ
- คัดแปลงข้อมูลที่อาจจะมีความไม่เข้ากันในแต่ละ format เพื่อให้เข้ากันได้เช่น บางครั้งการเก็บข้อมูลของบางเซิร์ฟเวอร์ จะเก็บข้อมูลเป็น mm/dd/yy แทนที่จะเป็น dd/mm/yyyy

- เชื่อมต่อกับคานาเบสและเขียนข้อมูลเข้าคานาเบส

ซึ่ง filter นั้นจะต้องสามารถอ่านและวิเคราะห์ไฟล์ข้อมูล ได้โดยอาศัยการตัดคำจากเครื่องหมายวรรคตอนต่าง ๆ ของ ล็อกไฟล์แปลงข้อมูลให้อยู่ในรูปแบบเดียวกัน จากนั้นก็เขียนข้อมูลเข้าดาต้าเบส ซึ่งการเขียนข้อมูลนี้จะต้องควบคุมการเชื่อมต่อและเขียนข้อมูลกับดาต้าเบสเพื่อไม่ให้ thread แต่ละ thread กินการทำงานของเครื่องมากเกินไปจนมีลักษณะคล้ายกับเครื่องค้าง โดยใช้ Class SqlConnection ของ .Net เป็นเครื่องมือในการทำงาน

ซึ่งในการทำงานของโปรแกรมนี้ อาจจะประกอบไปด้วย Collector ที่รับข้อมูลจากหลาย ๆ data source แล้วนำมารวบรวมกันเพื่อให้ได้เอาท์พุทตามที่ระบบต้องการจัดรูป ในกรณีนี้ ส่วน reader ที่เป็นตัวรวบรวมจะไม่มีบทบาทมากนัก โดยเราจะใช้ข้อได้เปรียบจากการเก็บข้อมูลลงในดาต้าเบส โดยนำข้อมูลของ collector ทั้งสองมา join ตารางกันเพื่อที่จะให้ได้ผลรวมที่ต้องการ



รูปที่ 3.6 แสดงลักษณะการเชื่อมโยงสำหรับการวิเคราะห์จากหลาย data source

ซึ่งเราสามารถสรุปขั้นตอนการทำงานของ collector ออกมาในลักษณะของ flowchart ได้ดังนี้

- เริ่มต้นจากการให้ผู้ใช้สร้าง collector สำหรับ data source แต่ละตัวและกำหนดการเชื่อมโยงระหว่าง collector แต่ละตัวในกรณีที่มี data source มากกว่า 1 ตัว

- กำหนดค่า parameters ต่าง ๆ สำหรับ collector ได้แก่ค่า IP address ของ data source, ชนิดของ log files username, password ที่ใช้ในการเชื่อมต่อกับ data source, กำหนดสถานะของ collector ว่าเป็นแบบ เดี่ยวหรือแบบกลุ่ม

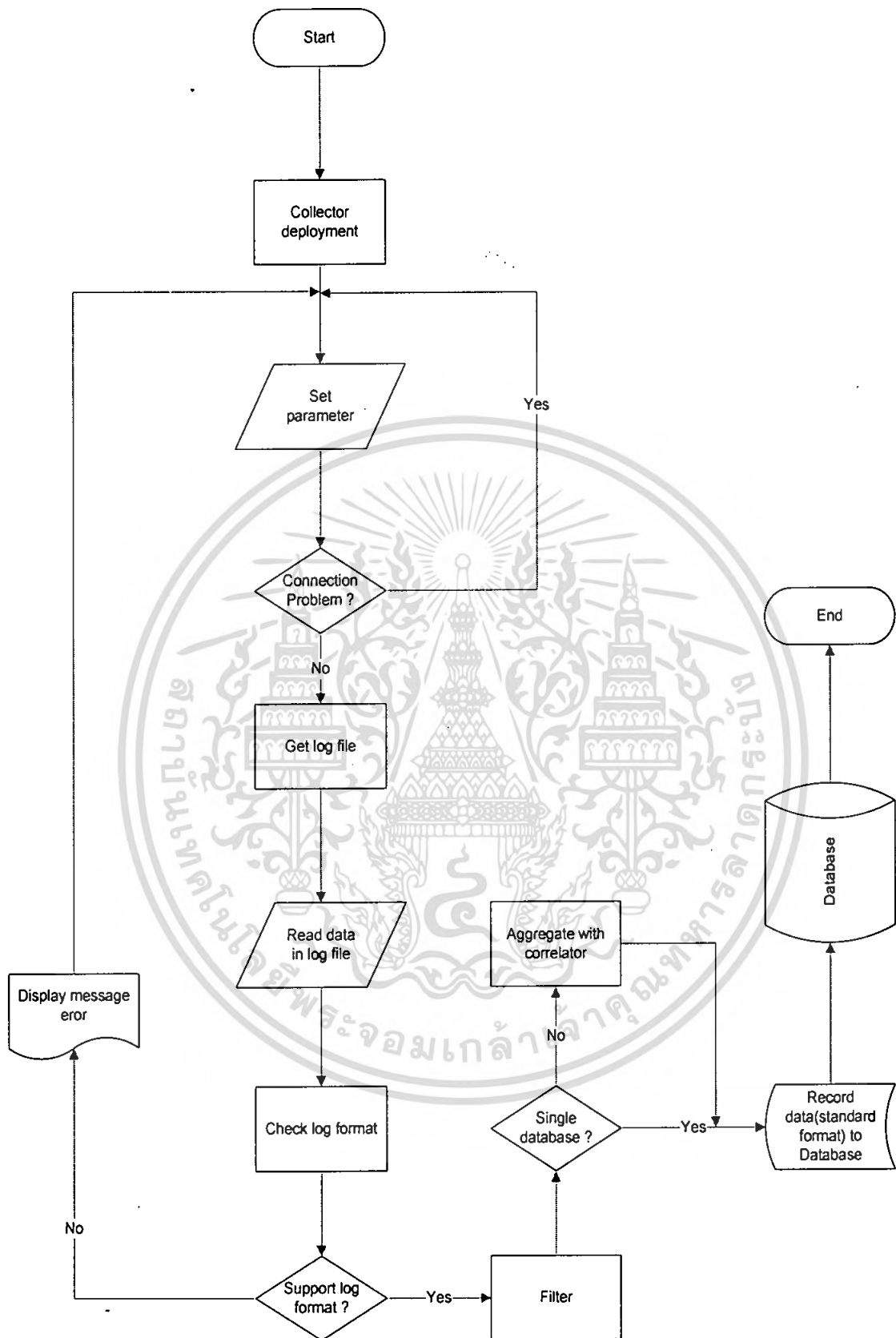
- ทำการรันเพื่อเชื่อมต่อกับ FTP server ของ เซิร์ฟเวอร์ ที่เราต้องการจะวิเคราะห์ ล็อกไฟล์ เพื่อทำการดึงลอกไฟล์ ส่วน reader จะส่ง request ไปยัง เซิร์ฟเวอร์ เพื่อรอการติดต่อกลับมา ถ้าสำเร็จก็จะทำการดึง ล็อกไฟล์นั้น ๆ มาจาก path ที่ได้ระบุไว้ก่อนหน้านี้

- ถ้าเกิดปัญหาในการเชื่อมต่อกับ FTP server เช่น connection failed หรือ username หรือ password มีข้อผิดพลาด program ก็จะได้รับ response แฉ่งกลับมาซึ่งโปรแกรมก็จะหยุดทำงานเพื่อให้ ผู้ใช้ได้ทำการแก้ไข

- ล็อกไฟล์ที่ดึงมานั้นจะเก็บไว้ในโฟลเดอร์ชั่วคราวจากนั้นจะให้ส่วน filter จัดการวิเคราะห์ไฟล์ แล้วนำไปเก็บไว้ใน database จากนั้นไฟล์ก็จะถูกลบทิ้งไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 แสดง flow chart การทำงานของ collector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2 Database

ในส่วนของระบบฐานข้อมูลใช้เก็บรายละเอียดของ ล็อกไฟล์ที่ผ่านการ Filter ให้อยู่ในรูปแบบมาตรฐาน โดยใช้ SQL express 2005 เป็นตัวจัดการข้อมูล เพื่อที่จะสามารถดึงข้อมูลไปวิเคราะห์ได้สะดวก โดยรายละเอียดของ Standard Format จะแสดงในตาราง 3.2

ตารางที่ 3.2 แสดงรายละเอียดของตาราง Standard Format

ชื่อฟิลด์	รายละเอียด
ID	Primary key ของตาราง Standard Format
IP-Client	IP Address ของ Client ที่ร้องขอบริการจาก Server
Date	วันเดือนปีที่ Server ได้รับ request
Time	เวลาที่ Server ได้รับ request
Status Code	Status Code ที่ Server ส่งกลับ ไปให้ Client
File	File ที่ Client request
Bytes_Received	จำนวน ไบต์ที่ Server ได้รับจาก Client
Bytes_Transfer	จำนวน ไบต์ที่ Server ส่งกลับ ไปให้ Client
Referer	เว็บเพจที่ให้ลิงค์เชื่อมต่อมาที่เว็บ ไซด์
Browser	Browser ของ Client ที่ขอ request
Os	Operating System ของ Client ที่ขอ request

จากตารางที่ 3.2 ได้แสดงรายละเอียดฟิลด์ในตาราง Standard Format ต่อจากนี้จะเป็นการแสดงความสัมพันธ์ว่าล็อกไฟล์ส่วนไหนที่ถูก filter ลงในแต่ละฟิลด์ในตาราง Standard Format โดยจะใช้ตัวอย่างล็อกไฟล์ของ Apache Combined Log Format

ฟิลด์ IP-Client

```
161.246.243.152 - - [30/Sep/2007:05:26:17 +0700] "GET /index/index.php HTTP/1.1" 200
25030 "http://www.kmitl.ac.th/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1;
.NET CLR 1.0.3705; .NET CLR 1.1.4322; Media Center PC 4.0)"
```

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟิลด์ Date และ Time

161.246.243.152 - - [30/Sep/2007:05:26:17 +0700] "GET /index/index.php HTTP/1.1" 200
 25030 "http://www.kmitl.ac.th/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1;
 .NET CLR 1.0.3705; .NET CLR 1.1.4322; Media Center PC 4.0)"

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

ฟิลด์ Status code

161.246.243.152 - - [30/Sep/2007:05:26:17 +0700] "GET /index/index.php HTTP/1.1" 200
 25030 "http://www.kmitl.ac.th/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1;
 .NET CLR 1.0.3705; .NET CLR 1.1.4322; Media Center PC 4.0)"

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

ฟิลด์ File

161.246.243.152 - - [30/Sep/2007:05:26:17 +0700] "GET /index/index.php HTTP/1.1" 200
 25030 "http://www.kmitl.ac.th/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1;
 .NET CLR 1.0.3705; .NET CLR 1.1.4322; Media Center PC 4.0)"

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

ฟิลด์ Byte_Transfer

161.246.243.152 - - [30/Sep/2007:05:26:17 +0700] "GET /index/index.php HTTP/1.1" 200
 25030 "http://www.kmitl.ac.th/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1;
 .NET CLR 1.0.3705; .NET CLR 1.1.4322; Media Center PC 4.0)" -

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

ไฟล์ Referrer

161.246.243.152 -- [30/Sep/2007:05:26:17 +0700] "GET /index/index.php HTTP/1.1" 200
 25030 "http://www.kmitl.ac.th/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1;
 .NET CLR 1.0.3705; .NET CLR 1.1.4322; Media Center PC 4.0)"

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

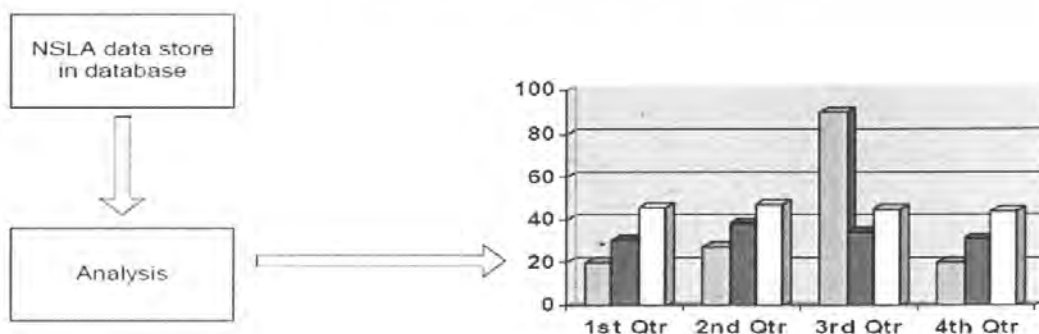
ไฟล์ Browser และ OS

161.246.243.152 -- [30/Sep/2007:05:26:17 +0700] "GET /index/index.php HTTP/1.1" 200
 25030 "http://www.kmitl.ac.th/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1;
 .NET CLR 1.0.3705; .NET CLR 1.1.4322; Media Center PC 4.0)"

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

3.4.3 Report

ในโปรแกรม NSAL จะมีตัว Generate Report ที่ทำหน้าที่ในการดึงข้อมูลที่เก็บในระบบฐานข้อมูลมาผ่านการ query ด้วยคำสั่ง SQL ในรูปแบบต่าง ๆ เพื่อให้ได้ข้อมูลที่ต้องการ จากนั้นนำข้อมูลมาใส่ในฟังก์ชันสร้างกราฟ หลายหลายรูปแบบ ที่เหมาะสมกับลักษณะกับข้อมูลที่จะนำเสนอ เช่น เมื่อเปรียบเทียบดูเปอร์เซ็นต์ ของ OS ที่เข้าใช้งานแล้วสร้างออกมาเป็นรูปปลงกลมก็น่าจะเหมาะสมกว่าเป็นต้น จากนั้นก็นำมารวมกันและสร้างเป็นสรุปผลการวิเคราะห์



Display Report

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.8 แสดงการทำงานของส่วน report
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.8 ในส่วนของการ Analysis ทำหน้าที่ดึงข้อมูลจากฐานข้อมูลนำมาวิเคราะห์และสร้างเป็นรายงาน โดยต่อจากนี้จะเป็นการแสดงความสัมพันธ์ว่าไฟล์ไหนใน Standard Format ที่ถูกดึงมาวิเคราะห์แล้วสร้างรายงาน

- รายงานตามเวลาที่ใช้
 - แสดง - จำนวนผู้เข้าเยี่ยมชม(ip ไม่ซ้ำ)
 - จำนวนครั้งการเข้าเยี่ยมชม
 - จำนวนเพจที่เรียกดู
 - แบนวิดท์

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

- รายงานตามประเภทของผู้เข้าใช้
 - แสดง - ผู้เข้าเยี่ยมชม โดยแสดงเป็น ip (10อันดับที่มากที่สุด)
 - จำนวนครั้งการเข้าเยี่ยมชม
 - จำนวนเพจที่เรียกดู
 - แบนวิดท์
 - เข้าเยี่ยมชมครั้งสุดท้ายเมื่อไร

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

- รายงานตามลักษณะการเยี่ยมชม
 - แสดง - browser ที่ใช้
 - OS ที่ใช้
 - จำนวนการเข้าเยี่ยมชม
 - เปอร์เซนต์ของจำนวนการเข้าเยี่ยมชม

ID	IP-Client	Date	Time	Status Code	File	Bytes_Received	Bytes_Transfer	Refferer	Browser	Os
----	-----------	------	------	-------------	------	----------------	----------------	----------	---------	----

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

ในบทนี้จะกล่าวถึงรายละเอียดต่าง ๆ ของโปรแกรม การติดตั้งและใช้งาน การทำงานของโปรแกรม ทั้งในส่วนของภารกิจ ล็อกไฟล์และในส่วนแสดงผลรายงาน รวมไปถึงรายละเอียดการทดลองใช้จริง ซึ่งได้ทดลองในการทดลองนี้ได้ทดสอบการวิเคราะห์ ล็อกไฟล์ที่ได้จาก เว็บเซิร์ฟเวอร์ของสำนักทะเบียน (<http://www.reg.kmitl.ac.th>) โดย ล็อกไฟล์ที่นำมาวิเคราะห์มีขนาดประมาณ 2.8 GB โดยเป็นบันทึกการทำงานในช่วงวันที่ 1 ตุลาคมถึงวันที่ 31 ตุลาคม โดยมีบางส่วนที่วิเคราะห์ออกมาได้จากตัวอย่างในส่วนของผลการแสดงรายงาน

4.1 การใช้งานในส่วนของ collector

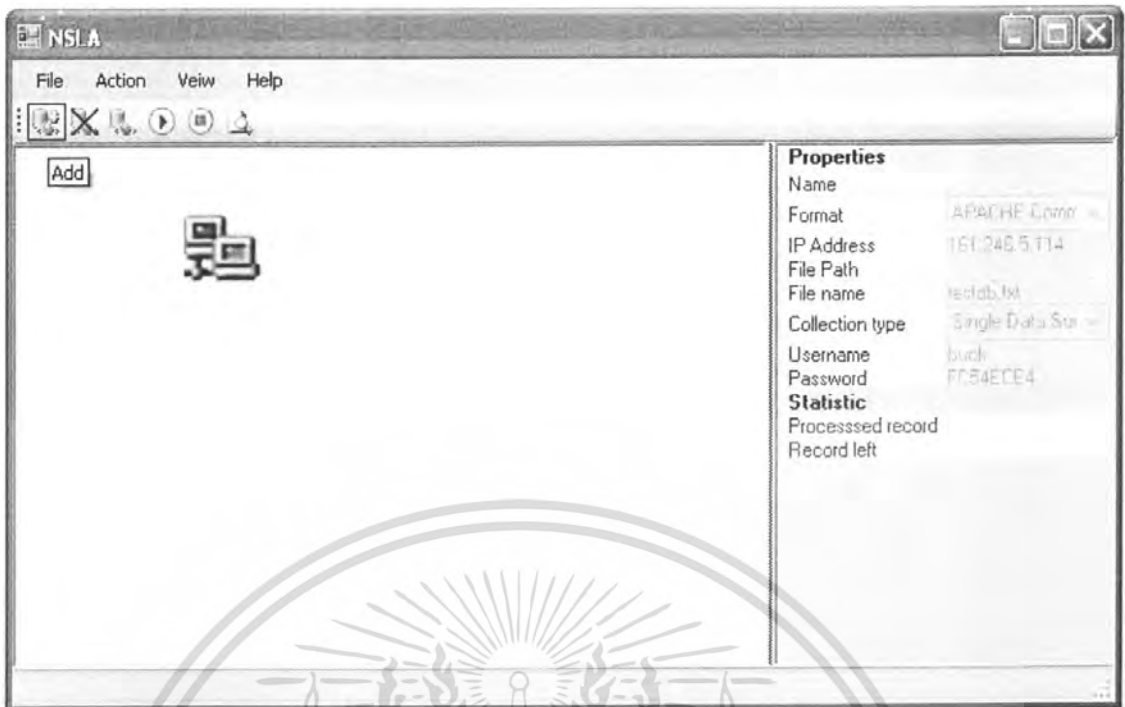
ในโปรแกรมที่สร้างขึ้นนี้ Collector จะเป็นส่วนหลักในการทำงานของโปรแกรม ซึ่งเมื่อเราจะเริ่มการทำงานนั้นก็จะต้องทำการสร้าง Collector ขึ้นมาและทำการกำหนดค่าเพื่อที่จะระบุถึงค่า parameter ต่าง ๆ ที่ใช้ในการทำงาน ซึ่งมีขั้นตอนปฏิบัติดังนี้

4.1.1. ทำการสร้าง Collector โดยเลือกจากปุ่ม new Collector ซึ่งเป็นปุ่มแรกที่อยู่บน tool bar



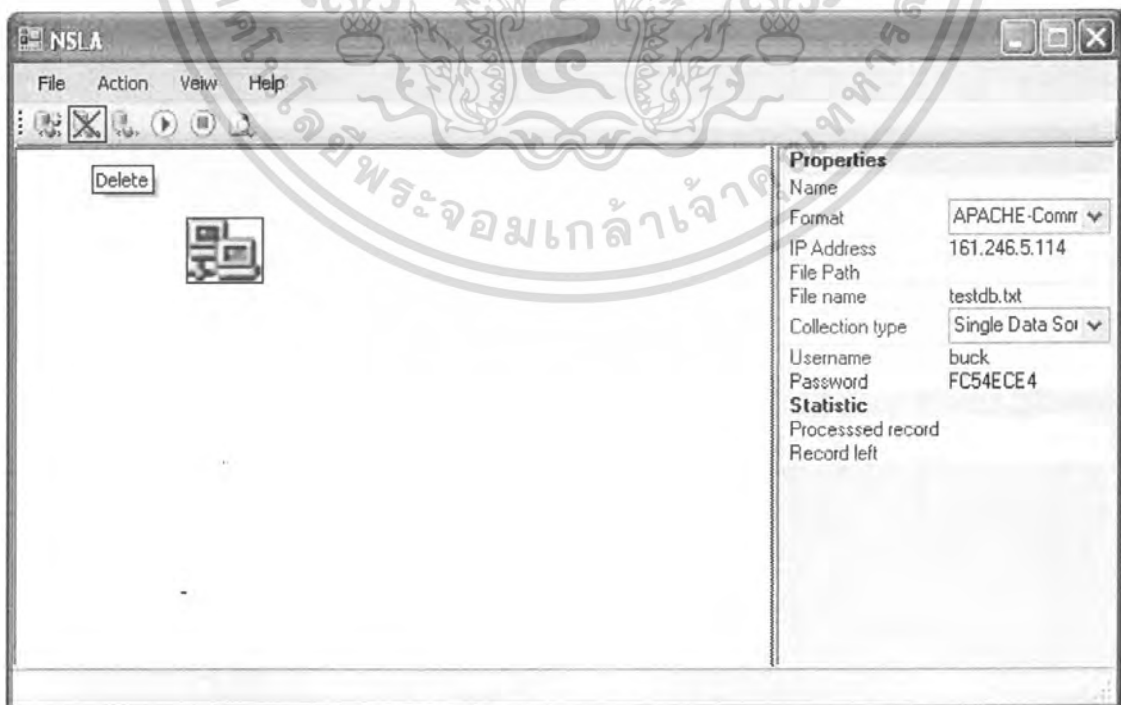
รูปที่ 4.1 แสดงการเลือกปุ่ม Add

หลังจากที่ click ปุ่ม Add แล้วจะมี Collector ปรากฏขึ้นมาดังที่ปรากฏในรูป 4.2



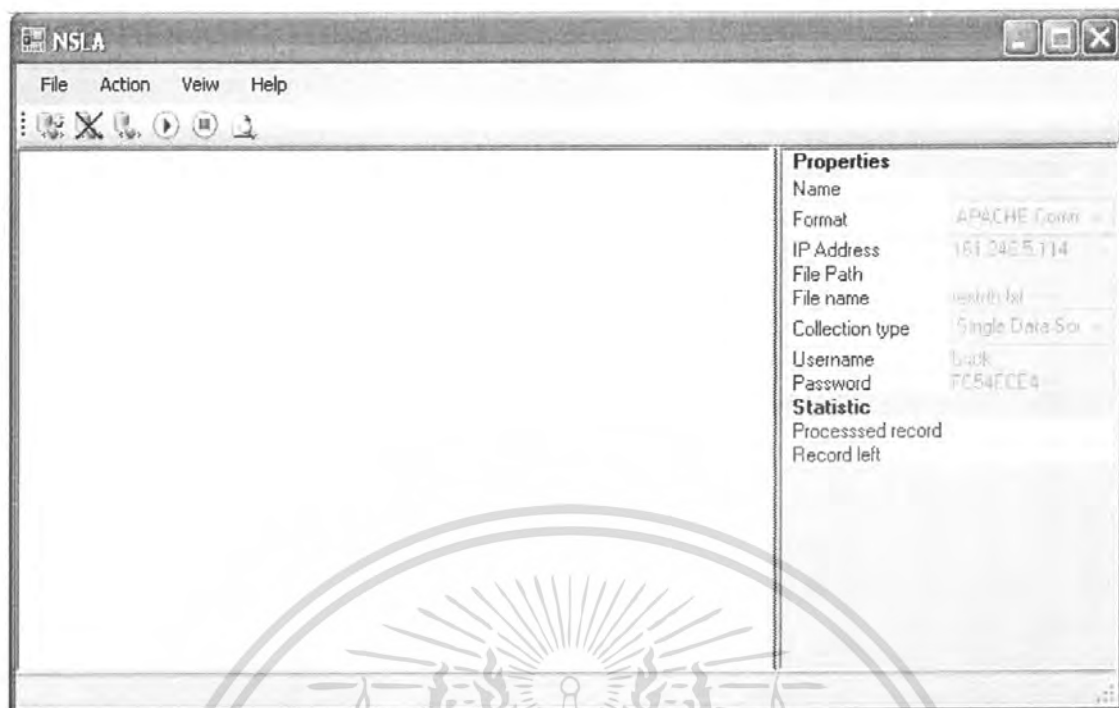
รูปที่ 4.2 แสดงผลหลังจากการคลิกปุ่ม Add

4.1.2 ลบ Collector โดยถ้า collector ที่สร้างขึ้นมาไม่เป็นที่ต้องการสามารถลบทิ้งได้โดยการเลือกที่ collector ตัวนั้นแล้วเลือกปุ่ม delete collector ซึ่งเป็นปุ่มที่สองที่อยู่บน tool bar collector ที่ถูกเลือกจะหายไป ดังที่แสดงให้เห็นในรูปที่ 4.4



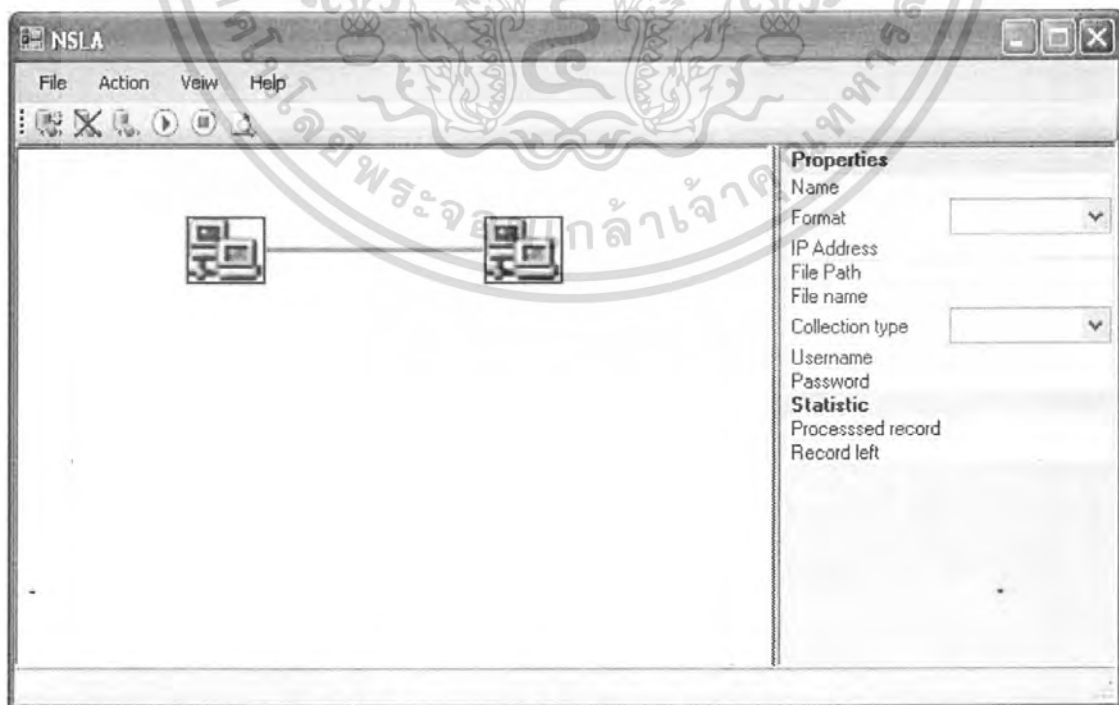
รูปที่ 4.3 แสดงการใช้ปุ่ม delete

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงผลลัพธ์ หลังจากการคลิกปุ่ม Delete

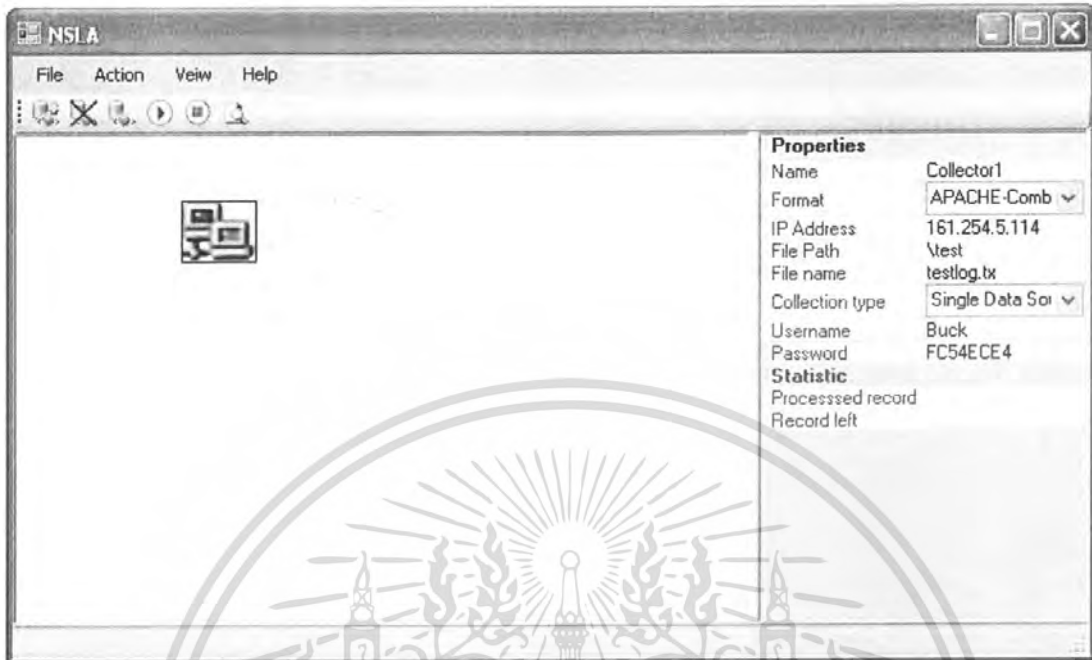
4.1.3 สามารถเชื่อม Collector เข้าหากันได้ โดยทำการเลือกที่ collector ตัวใดตัวหนึ่งแล้วลากเมาส์ไปที่ collector อีกตัว จะเป็นการกำหนดให้ collector ทั้งสองตัวทำงานร่วมกัน และจะทำการเชื่อมต่อ collector มากกว่า 2 ตัวขึ้นไปก็ทำได้เช่นกัน



รูปที่ 4.5 แสดงการเชื่อมต่อระหว่าง Collector

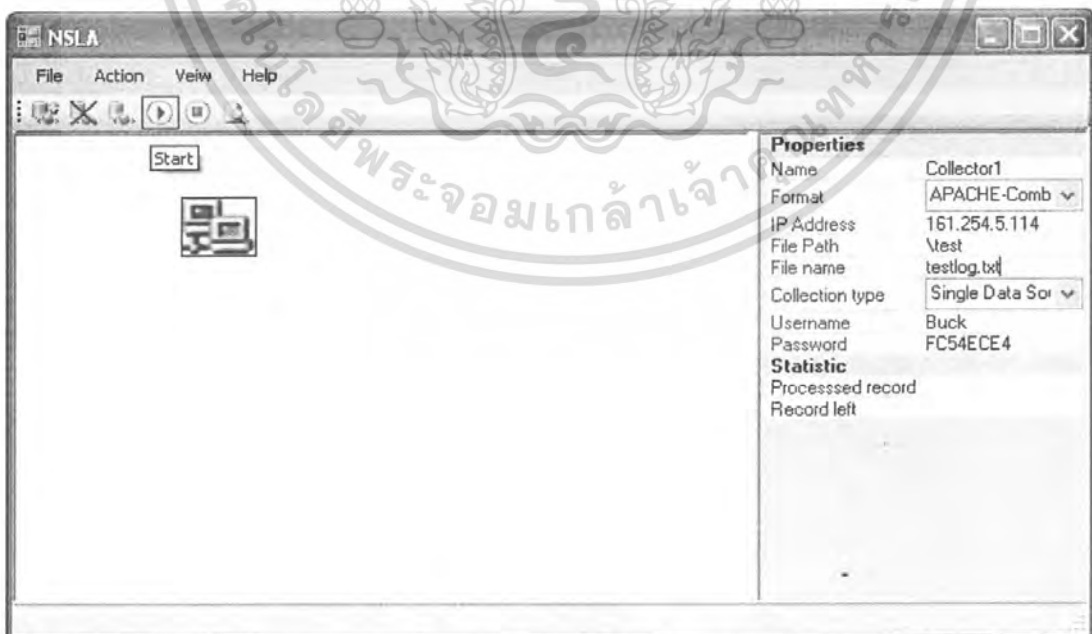
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.4 ทำการกำหนดค่า parameter ของ ล็อกไฟล์ที่ต้องการจะนำมาวิเคราะห์



รูปที่ 4.6 แสดงการกำหนดค่า parameter สำหรับ Collector แต่ละตัว

4.1.5 เลือกปุ่ม **start** ซึ่งเป็นปุ่มลำดับที่สี่บน tool bar เพื่อเริ่มการดึงล็อกไฟล์และเขียนข้อมูลลงบนดาต้าเบส

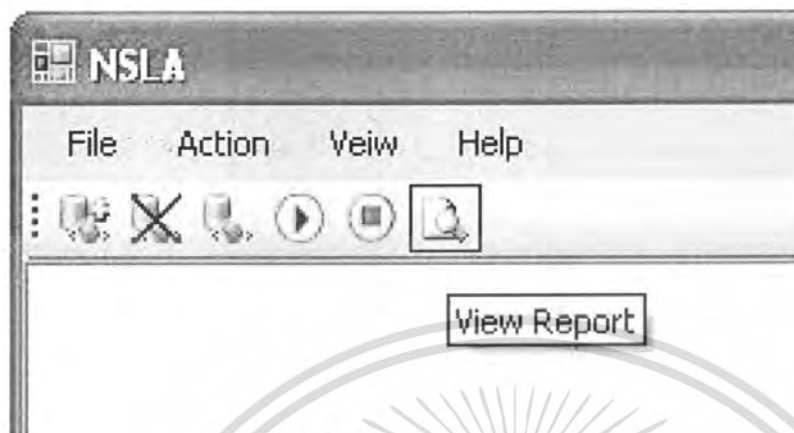


รูปที่ 4.7 แสดงการเลือกปุ่ม Start

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การ view Report

จากหน้าจอ โปรแกรมกดเลือก ไอคอน View Report ดังรูป



รูปที่ 4.8 แสดงการเลือกปุ่ม View report

จะปรากฏหน้าต่าง Report ขึ้นมาดังรูปที่ 4.8 และ 4.9 โดยจะ แบ่งออกเป็นสองส่วน



รูปที่ 4.9 แสดงผลของ Report ส่วนที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Summary Monthly History Days of Month Days of Week Host Connect to site from Browsers OS Robots Search word

ส่วนที่ 2

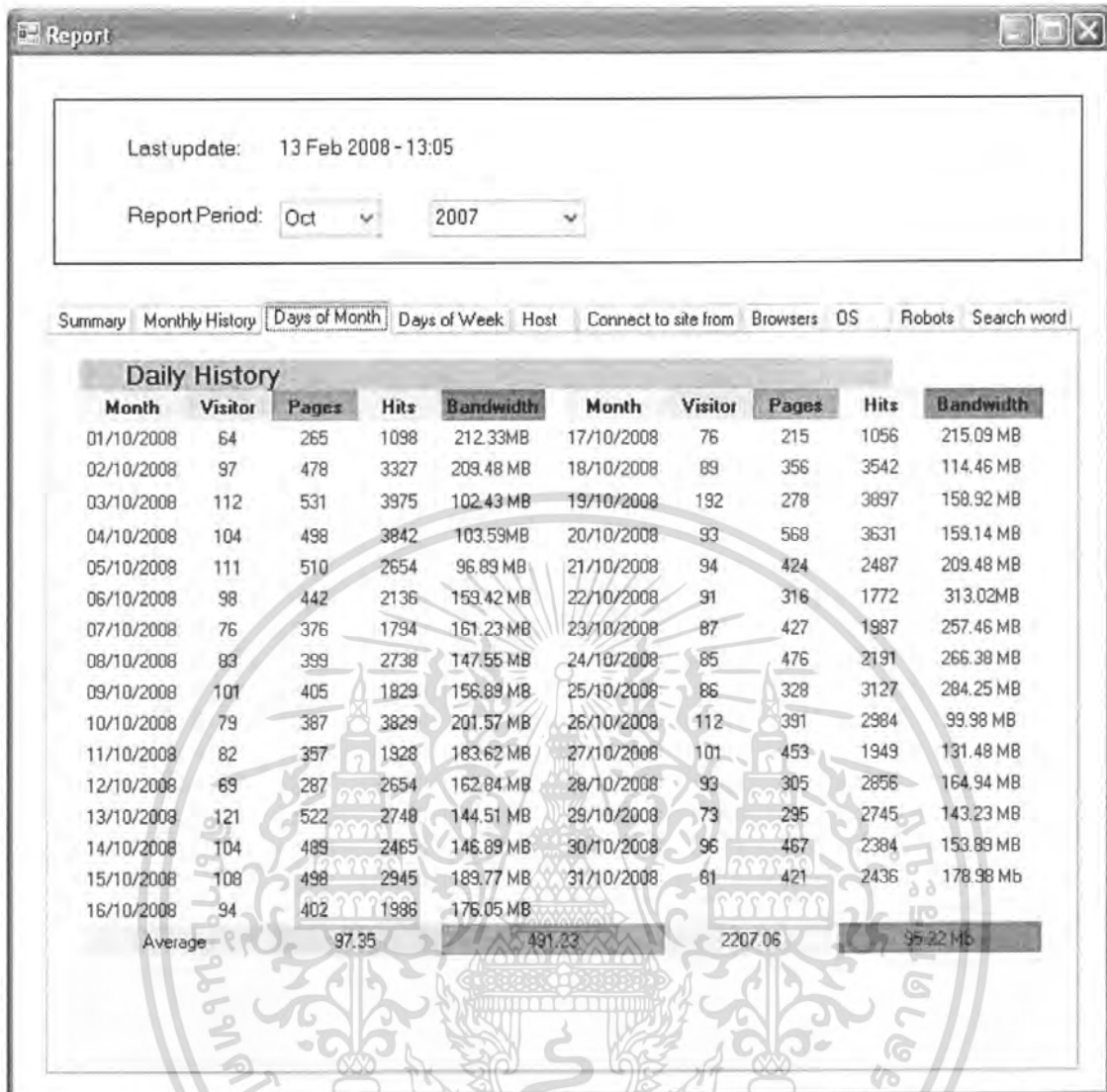
Summary				
Report period:	Month sep 2007			
First Visit:	01 Oct 2007			
Last Visit:	30 Oct 2007			
	Visitor	Pages	Hits	Bandwidth
Traffic	1775	15228	68419	2.88 GB

รูปที่ 4.10 แสดงผลของ Report ส่วนที่ 2

1. ส่วนที่หนึ่งในบรรทัดแรกจะเป็นการแสดงวันและเวลาที่ทำการดาวน์โหลดไฟล์มาครั้งล่าสุด ในบรรทัดที่สองจะเป็นการกำหนดช่วงระยะเวลาที่ต้องการทำการวิเคราะห์และแสดงผลออกมา โดยการเลือกเดือนและปี

2. ส่วนที่สองจะเป็นส่วนที่แสดงรายงานออกมา โดยจะมีแท็บให้เลือกว่าจะดูรายงานที่วิเคราะห์ตามเงื่อนไขต่างๆ โดยจากรูปที่ 4.10เป็นการแสดงรายงานผลสรุปของเดือนตุลาคม ปี 2007

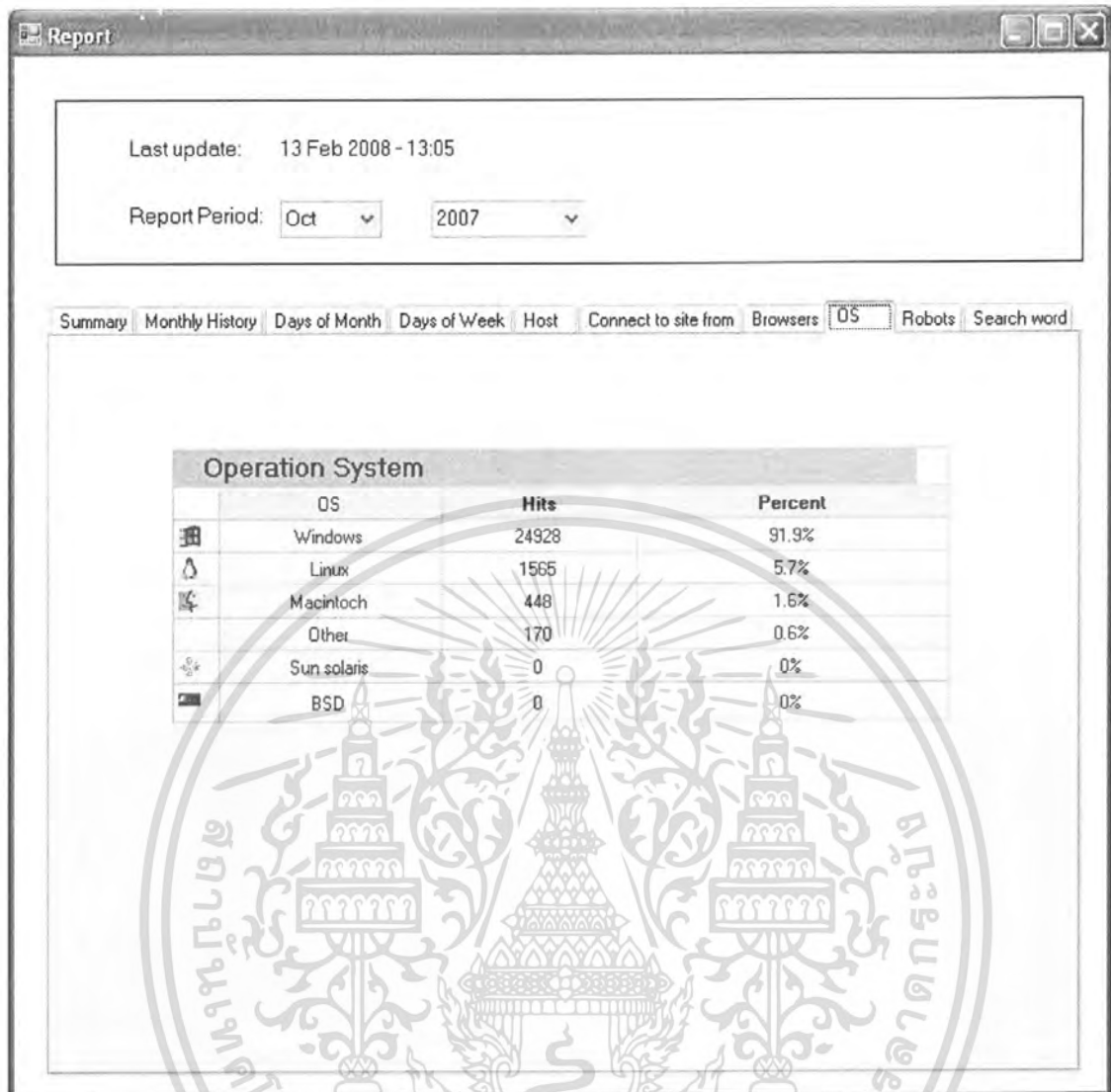
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 แสดงผล report ตามวันในแต่ละเดือน

จากรูปที่ 4.11 เป็นการแสดงรายงานที่วิเคราะห์ในแต่ละวันของเดือนที่เลือก ออกมาโดยแสดงเป็นจำนวนผู้เข้าชม , จำนวนหน้าที่มีการร้องขอ , จำนวนครั้งที่พบในล็อกไฟล์ และแบนด์วิธที่ใช้ในการตอบสนองความต้องการของ client

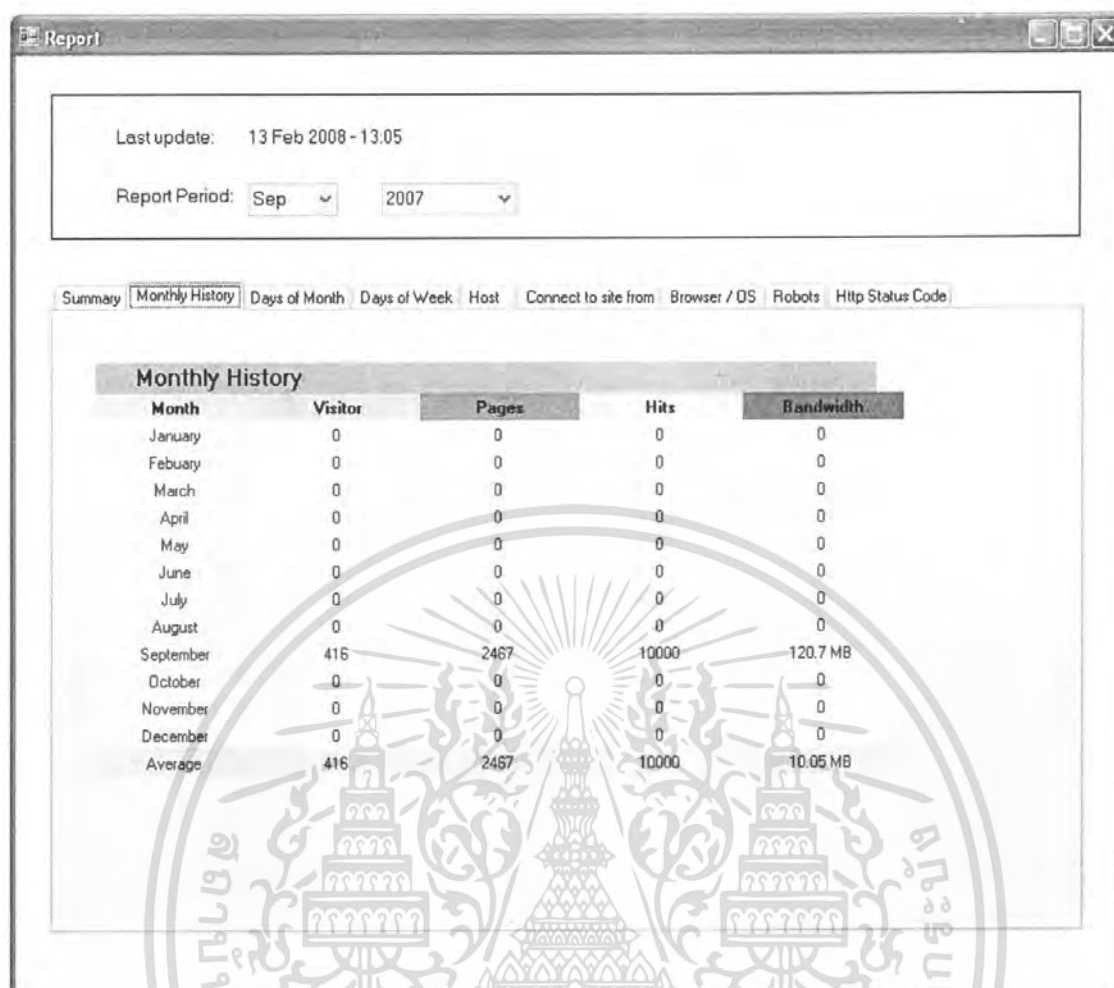
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 แสดงผล report ตาม OS

จากรูปที่ 4.12 เป็นการแสดงรายงานที่วิเคราะห์ประเภทของระบบปฏิบัติการของผู้ที่เข้ามาใช้บริการ โดยจะแสดงเป็นจำนวนครั้งที่พบในล็อกไฟล์ และเปอร์เซ็นต์เปรียบเทียบกับจำนวนครั้งทั้งหมด

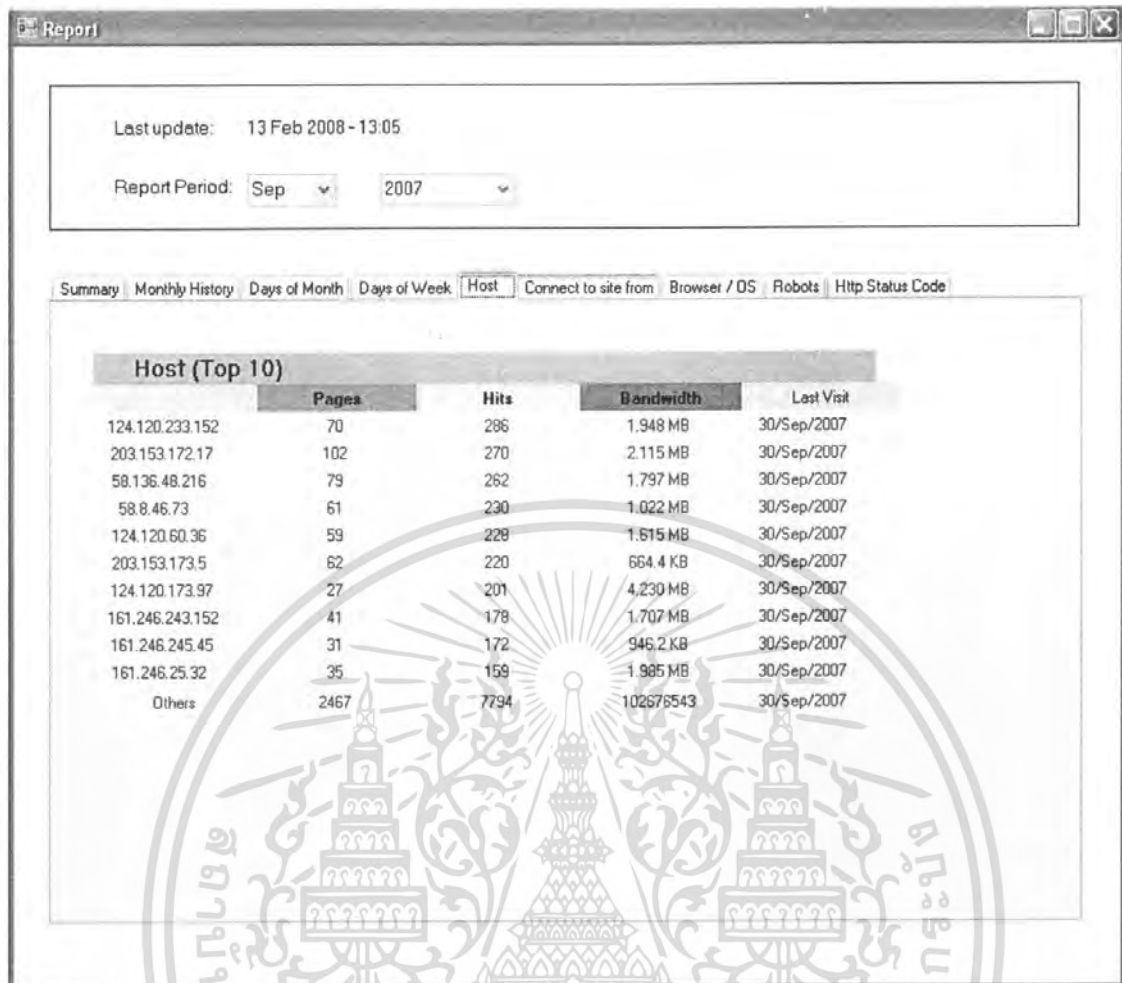
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 แสดงผล report ของแต่ละเดือน

จากรูปที่ 4.13 เป็นการแสดงรายงานที่วิเคราะห์ในแต่เดือน ออกมาโดยแสดงเป็นจำนวนผู้เข้าชม , จำนวนหน้าที่มีการร้องขอ , จำนวนครั้งที่พบในล็อกไฟล์ และ แบนด์วิธที่ใช้ในการตอบสนองความต้องการของ client

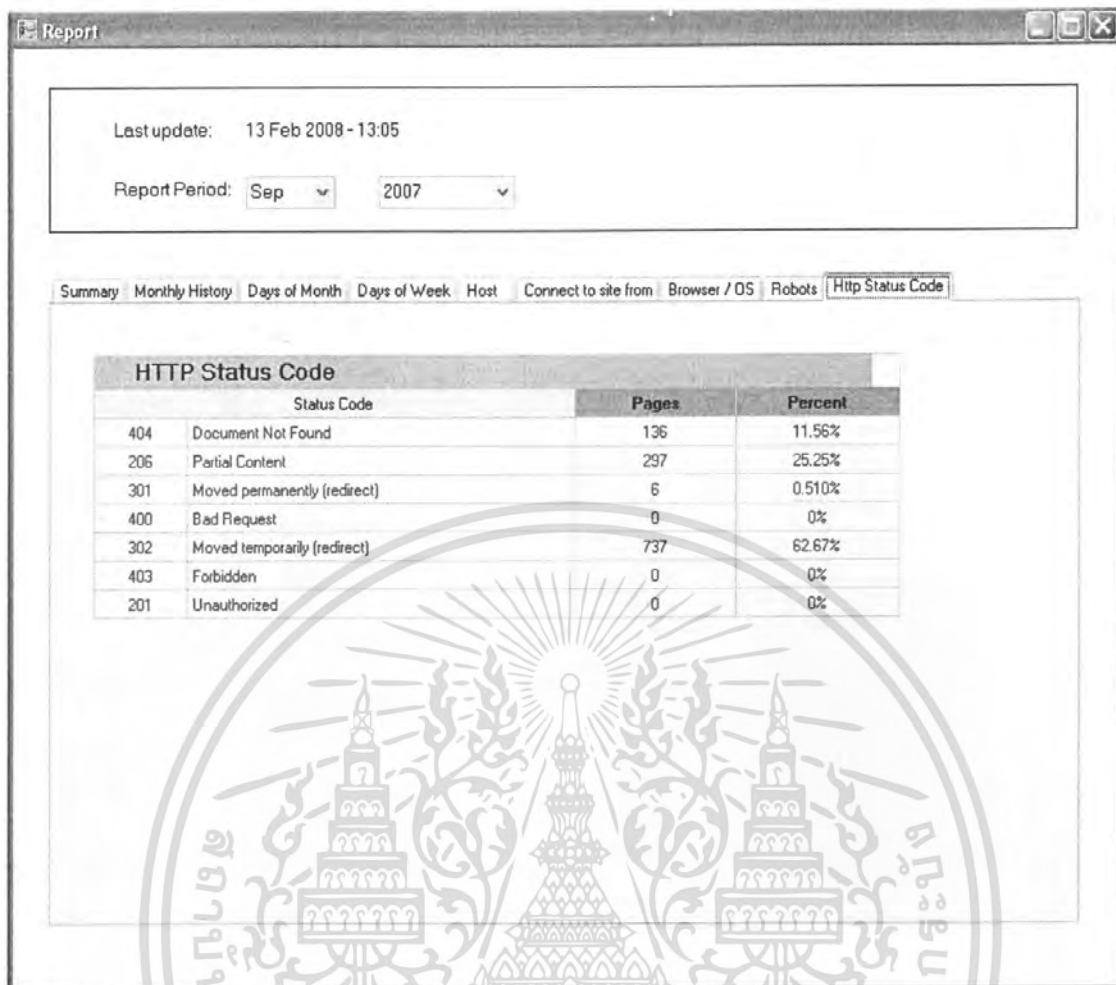
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 แสดงผล report ตาม ip ที่เข้ามาใช้

จากรูปที่ 4.14 เป็นการแสดงรายงานที่วิเคราะห์ตาม ip ของโฮสต์ที่เข้ามาใช้ ออกมาโดยแสดงเป็น จำนวนหน้าที่มีการร้องขอ , จำนวนครั้งที่พบในล็อกไฟล์ , แบนด์วิธที่ใช้ในการตอบสนองความต้องการของ client , และวันที่เข้าใช้ล่าสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 แสดงผล report ตาม HTTP Status Code

จากรูปที่ 4.15 เป็นการแสดงรายงานที่วิเคราะห์จากผลลัพธ์การเข้าชมเว็บไซต์ที่ผิดพลาดทั้งหมด โดยแสดงค่าจำนวนหน้าที่มีการร้องขอ และแสดงสัดส่วนทั้งหมดเป็นร้อยละ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และสรุป

5.1 บทสรุป

ในการทดลองการทำงานของ โปรแกรม Network Log Analyzer ทั้งการ จัดการข้อมูลลงฐานข้อมูลเพิ่มข้อมูล ลบข้อมูล และแสดงผลรายงานนั้น ได้ผลการทดลองที่น่าพอใจเป็นอย่างมาก โดยผลลัพธ์ที่ได้วิเคราะห์ออกมามีความหลากหลาย และเมื่อเทียบกับ โปรแกรมประเภทเดียวกันที่อยู่ในท้องตลาด กับว่ามี feature ที่ใกล้เคียงกันรวมถึงประสิทธิภาพและความเร็วในการทำงาน

5.2 วิจารณ์สิ่งที่ได้จากโครงการ

เทคโนโลยีที่นำมาใช้ในการพัฒนาของ โครงการในระบบนี้ ประกอบไปด้วย การพัฒนาโดยโปรแกรมด้วยวิธีแบบ OOP ทำให้สามารถนำโปรแกรมไปพัฒนาต่อได้โดยง่าย และการวิเคราะห์สื่อของเครื่องข่ายนั้นก็สามารถเพิ่มประสิทธิภาพของการทำงานของเซิร์ฟเวอร์ได้

5.3 ปัญหาอุปสรรคและแนวทางแก้ไข

1. ข้อจำกัดของสื่อไฟล์ที่เป็น ไฟล์ที่ไม่ค่อยเป็นที่นิยมเปิดเผย ทำให้ไฟล์ที่หามาได้มีข้อด้อยทางด้านความหลากหลายในการใช้งาน
2. ในบางกรณี การตั้ง เซิร์ฟเวอร์ อาจจะใช้การสร้างสื่อไฟล์ที่เป็นรูปแบบเฉพาะของเจ้าของเซิร์ฟเวอร์ เอง ทำให้ โปรแกรมไม่สามารถรองรับได้
3. การพัฒนาของตัว crawler สำหรับ search engine และ worm บางตัวพัฒนาไปเร็วมากและมีรูปแบบที่ซับซ้อนมากขึ้นทำให้ยากต่อการวิเคราะห์

5.4 แนวทางการพัฒนาต่อ

1. พัฒนาให้มีความยืดหยุ่นของ โปรแกรมมากขึ้นเช่น ตัวกรองข้อมูลของสื่อไฟล์สามารถให้ผู้ใช้กำหนดเพื่อให้เข้ากับสื่อไฟล์นั้น ๆ
2. พัฒนาโปรแกรมให้ยืดหยุ่นทางการแสดงผล เช่นสามารถดูการแสดงผลแบบเงื่อนไขเช่น IP

ที่มี hits มากกว่า 2000 ในหนึ่งวัน เป็นต้น เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. พัฒนาให้มีการเรียนรู้รูปแบบการเข้าใช้ของผู้ใช้ในเซิร์ฟเวอร์นั้นๆ เพื่อสามารถตรวจจับการเข้าใช้ที่ผิดปกติได้
4. พัฒนาให้ทำการเชื่อมต่อกับ เซิร์ฟเวอร์ ด้วยโปรโตคอลที่หลากหลายขึ้น เช่น HTML SMTP ฯลฯ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Microsoft MSDN. 2007. **FtpWebRequest Class (System.net)** [online]

Available: <http://msdn2.microsoft.com/en-us/library/system.net.ftpwebrequest.aspx>.

AWStats. 2007. **AWStats official documentation** [online]

Available: <http://awstats.sourceforge.net/docs/index.html>.

Analog. 2007. **Analog complete documentation version 6.0** [online]

Available: <http://www.analog.cx/docs/Readme.html>.

Webalizer. 2007. **Webalizer** [online]

Available: <http://www.mrunix.net/webalizer/download.html>.

Apache. 2007. **HTTP Server Version 2.2 Log Files** [Online]

Available :<http://httpd.apache.org/docs/2.2/logs.html>.

IBM. 2007. **Log File Format** [Online]

Available:http://publib.boulder.ibm.com/tividd/td/ITWSA/ITWSA_info45/en_US/HTML/guide/c-logs.html#combined.

Windows Server 2003 TechCenter. 2007. **W3C Extended Log File Examples** [Online]

Available:<http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/ffdd7079-47be-4277-921f-7a3a6e610dcb.mspx?mfr=true>.