ระบบแสดงตัวอย่างข้อมูลจากเว็บเพจสำหรับการเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำ

นายนิทัศน์ สุทวีปราโมชานนท์

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2545

ISBN 974-17-1664-8

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

WEB CONTENT PREVIEWING SUBSYSTEM FOR LOW BANDWIDTH CONNECTIVITY

Mr. Nitass Sutaveephamochanon

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science

Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University

Academic Year 2002

ISBN 974-17-1664-8

หัวข้อวิทยานิพนธ์	ระบบแสดงตัวอย่างข้อมูลจากเว็บเพจสำหรับการเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำ
โดย	นาย นิทัศน์ สุทวีปราโมชานนท์
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษา	อาจารย์ ดร. ยรรยง เต็งอำนวย

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

…….. คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร. สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

……ประธานกรรมการ
(อาจารย์ จารุมาตร ปิ่นทอง)

…… อาจารย์ที่ปรึกษา
(อาจารย์ ดร. ยรรยง เต็งอำนวย)

…… กรรมการ
(อาจารย์ ดร. ณัฐวุฒิ หนูไพโรจน์)

…… กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. ทวิตีย์ เสนีวงศ์ ณ อยุธยา)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

นิทัศน์ สุทวีปราโมชานนท์ : ระบบแสดงตัวอย่างข้อมูลจากเว็บเพจสำหรับการเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำ. (WEB CONTENT PREVIEWING SUBSYSTEM FOR LOW BANDWIDTH CONNECTIVITY) อ. ที่ปรึกษา : อาจารย์ ดร. ยรรยง เต็งอำนวย, 50 หน้า. ISBN 974-17-1664-8.

อินเทอร์เน็ตเป็นแหล่งข้อมูลขนาดใหญ่สำหรับการศึกษา ค้นคว้าต่าง ๆ การใช้งานอินเทอร์เน็ตโดยใช้การเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำทำให้ไม่สามารถใช้งานอินเทอร์เน็ตได้อย่างมีประสิทธิภาพ การแก้ปัญหาโดยการเพิ่มความกว้างช่องสัญญาณมีค่าใช้จ่ายที่สูง และมีข้อจำกัดบางประการ ในการวิจัยนี้จึงได้นำเสนอระบบแสดงตัวอย่างข้อมูลจากเว็บเพจขึ้นเพื่อช่วยให้การใช้งานอินเทอร์เน็ตที่มีความกว้างช่องสัญญาณต่ำสามารถทำได้สะดวก รวดเร็ว และมีประสิทธิภาพมากขึ้น อีกทั้งยังเป็นการลดปริมาณข้อมูลที่รับส่งกันในเครือข่ายอินเทอร์เน็ตได้อีกด้วย

การวิจัยนี้ได้นำเสนอระบบแสดงตัวอย่างข้อมูลภาพ และภาพเคลื่อนไหว โดยจะทำการตรวจสอบความกว้างช่องสัญญาณระหว่างเว็บเซิร์ฟเวอร์ และไคลเอนต์ก่อนการส่งข้อมูล การแสดงตัวอย่างข้อมูลจะพิจารณาจากความกว้างช่องสัญญาณ และขนาดของข้อมูลที่ไคลเอนต์ทำการร้องขอ ถ้าค่าที่คำนวณได้น้อยกว่าอัตราส่วนที่กำหนดไว้ระบบจะส่งตัวอย่างข้อมูลไปให้แก่ไคลเอนต์แทนโดยอัตโนมัติ เมื่อไคลเอนต์พบข้อมูลที่ต้องการแล้วสามารถทำการร้องขอข้อมูลในรูปแบบที่สมบูรณ์จากระบบได้ ในการวิจัยได้เลือกใช้ไฟล์เจเพ็ก และเอ็มเพ็ก ในการทดสอบ โดยเปรียบเทียบเวลาการแสดงตัวอย่างด้วยวิธีต่าง ๆ จากผลการทดสอบสามารถสรุปได้ว่า วิธีการแสดงตัวอย่างข้อมูลภาพที่เหมาะสม คือ การตัดขนาดของภาพแล้วทำการลดคุณภาพของภาพลง วิธีการแสดงตัวอย่างข้อมูลภาพเคลื่อนไหวที่เหมาะสม คือ การตัดช่วงของภาพเคลื่อนไหวแล้วลดขนาดของกรอบภาพเคลื่อนไหวลง สำหรับการตรวจสอบความกว้างช่องสัญญาณ และบริการการแสดงตัวอย่างข้อมูล จากการทดสอบสรุปได้ว่า วิธีการตรวจสอบความกว้างช่องสัญญาณที่หมาะสม คือ การใช้แอปเพล็ต และการพัฒนาบริการการแสดงตัวอย่างข้อมูลที่เหมาะสม คือ การใช้เว็บเซิร์ฟเวอร์เอพีไอ ในการพัฒนา

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต.....
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่ออาจารย์ที่ปรึกษา.....
ปีการศึกษา 2545

42703848 - 21 : MAJOR COMPUTER SCIENCE
KEY WORD: PREVIEW / BANDWIDTH / WEB SERVER / HTTP

NITASS SUTAVEEPHAMOCHANON : WEB CONTENT PREVIEWING SUBSYSTEM FOR LOW BANDWIDTH CONNECTIVITY. THESIS ADVISOR : DR. YUNYONG TENG - AMNUAY, 50 pp. ISBN 974-17-1664-8.

The Internet is a huge resource for researching and studying. Low bandwidth connection to the Internet might result in ineffective utilization. One of the solutions is to broaden the bandwidth; however, this solution is very expensive yet still has some other limitations. In this research, we introduce the web content previewing subsystem to allow us to utilize effectively the Internet with low bandwidth. Also, this new system will decrease amount of information packages in Internet traffic.

This research demonstrates how the subsystem works for image and video data. First, it will detect an available bandwidth between web server and client before sending an actual package. Then, the system will determine if the preview process will be launched. The process will be launched in the situation when a low bandwidth is detected but the requested data is large. Otherwise, the data will be sent to the client without employing the preview process. This research uses the JPEG file format for image data and MPEG file format for video data for studying. We found that the most effective way for previewing the image data is to crop and decrease the quality of the image. For video data, we have found that by skipping and decreasing the size of the video data the preview process will be improved. Finally, the conclusions of this study are that the best way to detect the bandwidth is to use an applet and the most appropriate web server for the preview process is an web server API.

Department Computer Engineering Student's signature.....
Field of study Computer Science Advisor's signature.....
Academic year 2002

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของอาจารย์ ดร. ยรรยง เต็งอำนวย อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้เอาใจใส่ ให้คำแนะนำ และข้อคิดเห็นต่าง ๆ ในการวิจัยด้วยดีตลอด รวมทั้งพี่ ๆ น้อง ๆ ใน ISEL LAB ทุกคนที่ได้ช่วยออกความคิดเห็นในการแก้ไขปัญหาที่เกิดขึ้นระหว่างการวิจัย

ท้ายนี้ ผู้วิจัยใคร่ขอกราบขอบพระคุณ บิดา – มารดา ซึ่งสนับสนุนในด้านการเงิน และให้กำลังใจแก่ผู้วิจัยเสมอมาจนสำเร็จการศึกษา

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ฌ

สารบัญภาพ ..... ญ

บทที่

1 บทนำ 1

1.1 ความเป็นมา และปัญหา 1

1.2 วัตถุประสงค์ของการวิจัย 1

1.3 ขอบเขตของการวิจัย 1

1.4 แนวคิดการดำเนินการวิจัย 2

1.5 วิธีดำเนินการวิจัย 3

1.6 ประโยชน์ที่คาดว่าจะได้รับ 3

2 ทฤษฎี และการวิจัยที่เกี่ยวข้อง 4

2.1 ทฤษฎีที่เกี่ยวข้อง 4

2.1.1 เอชทีทีพี 4

2.1.2 อาปาเช่ 7

2.1.3 แอปเพล็ต 9

2.1.4 การบีบอัดคงสัญญาณหลัก 10

2.1.5 ข้อมูลมัลติมีเดียในอินเตอร์เน็ต 11

2.2 การวิจัยที่เกี่ยวข้อง 13

2.2.1 Previews and Overviews in a Prototype for the Library of Congress National Digital Library 13

2.2.2 Web Cache Management Techniques for Images และ Design and Implementation of a Soft Caching Proxy 13

2.2.3 Web Server Workload Characterization 13

2.2.4 An Investigation of Documents from the World Wide Web 13

2.2.5 การวิเคราะห์ความสัมพันธ์ระหว่างลักษณะของยูอาร์แอลกับการใช้งานเว็บแคช 14

สารบัญ (ต่อ)

หน้า

3 การออกแบบ และพัฒนาระบบ 15
3.1 การตรวจสอบความกว้างช่องสัญญาณ 15
3.1.1 โดยการ Ping 15
3.1.2 โดยการใช้แอปเพล็ต 16
3.2 การเก็บข้อมูลความกว้างช่องสัญญาณ 16
3.3 ค่าความล่าช้า 18
3.4 การแสดงตัวอย่างข้อมูล 18
3.4.1 ข้อมูลภาพ 18
3.4.2 ข้อมูลภาพเคลื่อนไหว 29
3.5 อินเตอร์เฟส 37
3.6 เครื่องมือที่ใช้ในการวิจัย 38
4 ผลการวิจัย 42
4.1 การตรวจสอบความกว้างช่องสัญญาณ 42
4.2 การเก็บข้อมูลความกว้างช่องสัญญาณ 42
4.3 ความล่าช้า 42
4.4 การแสดงตัวอย่างข้อมูล 42
4.5 ระบบแสดงตัวอย่างข้อมูลจากเว็บเพจ 43
4.6 ข้อมูลตัวอย่าง 43
4.7 สภาพแวดล้อมของเครื่องที่ใช้พัฒนา 44
4.8 การทดสอบระบบ 44
5 สรุปผลการวิจัย อภิปรายผล ปัญหาและข้อเสนอแนะ 46
5.1 สรุปผลการวิจัย 46
5.2 อภิปรายผลการวิจัย 46
5.3 ปัญหาในการวิจัย 47
5.4 ข้อเสนอแนะ 47
รายการอ้างอิง 48
ประวัติผู้เขียนวิทยานิพนธ์ 50

สารบัญตาราง

ตารางที่ หน้า

3.1 แสดงรายละเอียดของโพรโตคอลต่าง ๆ ที่ใช้งานโดย Net::Ping และ Net::Ping::External Perl Module 15

3.2 แสดงขนาดไฟล์ และเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพขนาดใหญ่ด้วยวิธีต่าง ๆ..... 20

3.3 แสดงขนาดไฟล์ และเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพขนาดใหญ่มากด้วยวิธีต่าง ๆ 21

3.4 แสดงขนาดไฟล์ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ..... 23

3.5 แสดงขนาดไฟล์ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ..... 24

3.6 แสดงขนาดไฟล์ และเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่ด้วยวิธีต่าง ๆ..... 30

3.7 แสดงขนาดไฟล์ และเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่มากด้วยวิธีต่าง ๆ..... 30

3.8 แสดงขนาดไฟล์ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ..... 32

3.9 แสดงขนาดไฟล์ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ..... 32

4.1 เปรียบเทียบเวลาที่ใช้ทั้งหมดในการแสดงตัวอย่างข้อมูล และแสดงข้อมูลแบบสมบูรณ์ของข้อมูลภาพ..... 45

4.2 เปรียบเทียบเวลาที่ใช้ทั้งหมดในการแสดงตัวอย่างข้อมูล และแสดงข้อมูลแบบสมบูรณ์ของข้อมูลภาพเคลื่อนไหว..... 45

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

รูปที่ หน้า

1.1 แผนภาพแสดงการทำงาน. 2

2.1 การทำงานของอาปาเช่ 7

2.2 ขั้นตอนต่าง ๆ ในระหว่างการวนลูปของอาปาเช่ 8

2.3 แสดงขั้นตอนการพัฒนาโปรแกรมภาษาจาวา 9

2.4 แสดงลักษณะการเขียนโปรแกรมเพียงครั้งเดียวแล้วทำงานบนเครื่องคอมพิวเตอร์ชนิดใด ๆ 9

2.5 แสดงองค์ประกอบของจาวาแพลตฟอร์ม 10

3.1 แสดงเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพด้วยวิธีต่าง ๆ 22

3.2 แสดงเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ 25

3.3 แสดงเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ 26

3.4 แสดงเวลาที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ 27

3.5 แสดงเวลาที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ 28

3.6 แสดงการตัดช่วงของภาพเคลื่อนไหวมา 25% 29

3.7 แสดงเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพเคลื่อนไหวด้วยวิธีต่าง ๆ 31

3.8 แสดงเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ 33

3.9 แสดงเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ 34

3.10 แสดงเวลาที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ 35

3.11 แสดงเวลาที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ 36

3.12 แสดงเว็บเพจที่แสดงตัวอย่างข้อมูลภาพ 37

3.13 แสดงเว็บเพจที่แสดงข้อมูลภาพในรูปแบบที่สมบูรณ์ 37

3.14 แสดงเว็บเพจที่แสดงตัวอย่างข้อมูลภาพเคลื่อนไหว 38

3.15 แสดงเว็บเพจที่แสดงข้อมูลภาพเคลื่อนไหวในรูปแบบที่สมบูรณ์ 38

3.16 แสดงขั้นตอนการทำงานของระบบ 39

4.1 แสดงการทดสอบระบบแสดงตัวอย่างข้อมูลจากเว็บเพจ 44

บทที่ 1
บทนำ

1.1. ความเป็นมา และปัญหา

ในระยะเวลาไม่กี่ปีที่ผ่านมานี้ ความนิยมในการใช้งานอินเทอร์เน็ต (Internet) ได้เพิ่มขึ้นอย่างรวดเร็ว ทั้งนี้เนื่องจากการที่เทคโนโลยีมีการพัฒนาขึ้นอย่างมาก ทำให้การใช้งานอินเทอร์เน็ตมีความหลากหลายมากยิ่งขึ้น ไม่ใช่มีแต่ข้อความเพียงอย่างเดียวเหมือนแต่ก่อน มีการใช้รูปภาพ เสียง และภาพเคลื่อนไหวต่าง ๆ เพิ่มเข้าไปอีกด้วย ทำให้การใช้งานอินเทอร์เน็ตมีความน่าสนใจมากยิ่งขึ้น ในปัจจุบันนี้อินเทอร์เน็ตเป็นเครือข่ายสาธารณะที่มีขนาดใหญ่ที่สุดในโลก มีแหล่งข้อมูลต่าง ๆ อยู่เป็นจำนวนมาก ผู้ใช้สามารถใช้เครื่องคอมพิวเตอร์จากที่ใด ๆ ก็ได้เชื่อมต่อเข้ากับเครือข่ายอินเทอร์เน็ตเพื่อค้นหาข้อมูลที่ต้องการได้อย่างสะดวก และรวดเร็ว แต่ในการเชื่อมต่อที่มีความกว้างช่องสัญญาณ (Bandwidth) ต่ำนั้น การค้นหาข้อมูลที่มีขนาดใหญ่ เช่น ข้อมูลรูปภาพ เสียง หรือภาพเคลื่อนไหวต่าง ๆ จะทำให้เกิดปัญหาในการรับส่งข้อมูลขึ้น เช่น ใช้เวลาในการรับส่งข้อมูลมากเกินไป ทำให้ผู้ใช้ไม่สามารถค้นหาข้อมูลที่ต้องการได้สะดวก

การแก้ปัญหาโดยการเพิ่มขนาดความกว้างช่องสัญญาณนั้นสามารถทำได้ยากเนื่องจากมีค่าใช้จ่ายที่สูง และมีข้อจำกัดบางประการ เช่น ในการเชื่อมต่อโดยใช้การหมุนโมเด็มผ่านระบบโทรศัพท์นั้นจะมีความกว้างช่องสัญญาณจำกัดที่ 56 กิโลบิตต่อวินาที (Kilobit per second: Kbps) เท่านั้น ดังนั้นการใช้ การแสดงตัวอย่าง (Preview) ข้อมูลเพื่อช่วยในการค้นหาข้อมูลที่ต้องการจึงเป็นทางเลือกหนึ่งในการแก้ปัญหาดังกล่าว และยังเป็นการลดปริมาณข้อมูลที่รับส่งกันบนเครือข่ายอินเทอร์เน็ตได้อีกด้วย โดยในงานวิจัยนี้ได้นำเสนอระบบแสดงตัวอย่างข้อมูลรูปภาพ หรือภาพเคลื่อนไหวที่มีขนาดใหญ่จากเว็บเพจเมื่อมีการร้องขอเว็บเพจจากผู้ขอใช้บริการที่มีความกว้างช่องสัญญาณในการเชื่อมต่อต่ำ

1.2. วัตถุประสงค์ของการวิจัย

เพื่อออกแบบ และพัฒนาระบบแสดงตัวอย่างข้อมูลที่มีขนาดใหญ่จากเว็บเพจเมื่อมีการร้องขอเว็บเพจจากผู้ขอใช้บริการที่มีความกว้างช่องสัญญาณในการเชื่อมต่อต่ำ

1.3. ขอบเขตของการวิจัย

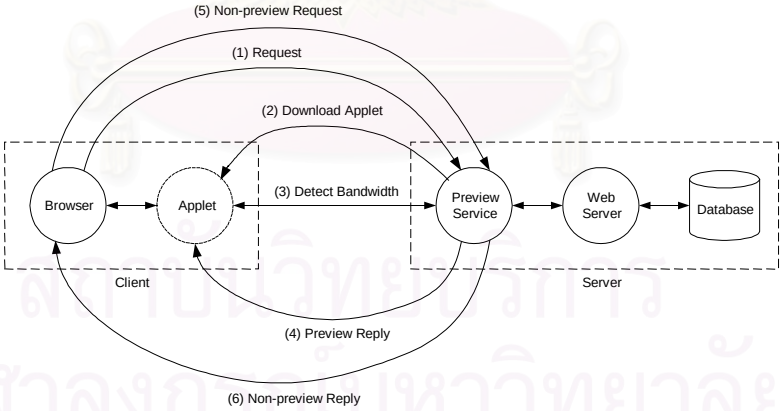
1. ทำบริการแสดงตัวอย่างข้อมูลสำหรับอาปาเช่ (Apache) บนยูนิกซ์แพลตฟอร์ม (UNIX Platform) เป็นอย่างน้อย
2. ทำแอปเพล็ต (Applet) สำหรับตรวจสอบขนาดความกว้างช่องสัญญาณในการเชื่อมต่อ
3. ในงานวิจัยนี้จะทำการแสดงตัวอย่างข้อมูลรูปภาพ และภาพเคลื่อนไหว เป็นอย่างน้อย
4. สามารถปรับ หรือเพิ่มชนิดของข้อมูลที่จะแสดงตัวอย่างได้สะดวก
5. การแสดงตัวอย่างข้อมูลใด ๆ จะใช้ซอฟต์แวร์ที่มีให้บนอินเทอร์เน็ต หรือหาได้ในลักษณะที่เป็นฟรีแวร์ (Freeware)
6. สามารถปรับค่าอัตราส่วนในการแสดงตัวอย่างข้อมูลได้สะดวก

1.4. แนวคิดการดำเนินการวิจัย

ทำบริการแสดงตัวอย่างข้อมูลให้กับเว็บเซิร์ฟเวอร์ โดยมีลักษณะโปร่งใส (Transparent) และอัตโนมัติ (Automatic) กับทุกเว็บเพจในเว็บเซิร์ฟเวอร์ โดยจะทำการตรวจสอบขนาดของข้อมูลในเว็บเพจ และขนาดความกว้างช่องสัญญาณในการเชื่อมต่อในขณะนั้น เพื่อจัดส่งข้อมูลในรูปแบบที่เหมาะสม และหลังจากที่ผู้ใช้พบข้อมูลที่ต้องการแล้วสามารถทำการร้องขอข้อมูลในรูปแบบที่สมบูรณ์ได้สะดวก

แนวทางหนึ่งในการทำระบบแสดงตัวอย่างข้อมูลนี้เป็นดังรูปที่ 1.1 โดยมีขั้นตอนการทำงานดังต่อไปนี้

1. ไคลเอนต์ทำการร้องขอเว็บเพจที่มีข้อมูลขนาดใหญ่จากเว็บเซิร์ฟเวอร์
2. เว็บเซิร์ฟเวอร์ได้รับการร้องขอเว็บเพจจากไคลเอนต์แล้ว จะทำการส่งแอปเพล็ตให้แก่ไคลเอนต์
3. ไคลเอนต์ได้รับแอปเพล็ตจากเว็บเซิร์ฟเวอร์แล้วจะทำการติดตั้งลงที่ไคลเอนต์ และใช้แอปเพล็ตในการติดต่อกับบริการแสดงตัวอย่างข้อมูลที่อยู่บนเว็บเซิร์ฟเวอร์เพื่อทำการตรวจสอบขนาดความกว้างช่องสัญญาณในการเชื่อมต่อในขณะนั้น
4. เว็บเซิร์ฟเวอร์นำข้อมูลที่ตรวจสอบได้ และขนาดของข้อมูลในเว็บเพจ มาทำการคำนวณและเปรียบเทียบกับค่าความล่าช้า (Latency) ที่ได้กำหนดไว้ ถ้าค่าที่ได้มีค่าน้อยกว่าค่าอัตราส่วนที่กำหนดไว้ก็จะทำการส่งข้อมูลในรูปแบบที่สมบูรณ์ แต่ถ้าค่าที่ได้มีค่ามากกว่าจะทำการลดปริมาณข้อมูลก่อนทำการส่งไปให้ไคลเอนต์
5. เมื่อผู้ใช้พบข้อมูลที่ต้องการแล้ว ไคลเอนต์จะทำการร้องขอข้อมูลที่สมบูรณ์จากเว็บเซิร์ฟเวอร์
6. เว็บเซิร์ฟเวอร์ได้รับการร้องขอข้อมูลที่สมบูรณ์จากไคลเอนต์แล้วจะทำการส่งเว็บเพจที่มีข้อมูลในรูปแบบที่สมบูรณ์ให้แก่ไคลเอนต์



รูปที่ 1.1 แผนภาพแสดงการทำงาน

1.5. วิธีดำเนินการวิจัย

1. ศึกษาการทำงานของอาปาเช่ และแอปเพล็ต
2. ศึกษา และวิเคราะห์การแสดงตัวอย่างข้อมูลแบบต่าง ๆ
3. กำหนดค่าอัตราส่วนที่จะทำการแสดงตัวอย่างข้อมูล
4. ออกแบบส่วนติดต่อผู้ใช้ (User Interface) ที่เหมาะสม เพื่อให้การแสดงตัวอย่างข้อมูลและการดูข้อมูลในรูปแบบที่สมบูรณ์รบกวนการทำงานปกติน้อยที่สุด
5. ออกแบบ และพัฒนาแอปเพล็ตสำหรับโปรแกรมบราวเซอร์
6. ออกแบบ และพัฒนาบริการแสดงตัวอย่างข้อมูลสำหรับอาปาเช่
7. ทำการทดสอบ และแก้ไขระบบ
8. จัดทำวิทยานิพนธ์

1.6. ประโยชน์ที่คาดว่าจะได้รับ

1. ลดปริมาณข้อมูล และเวลาที่ใช้ในการรับส่งข้อมูลในเครือข่ายอินเทอร์เน็ต
2. ผู้ใช้สามารถใช้เว็บได้รวดเร็วขึ้น เนื่องจากได้รับข้อมูลกลับมาเร็วแม้ว่าจะใช้การเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำก็ตาม
3. ลดความต้องการในการทำตัวอย่างข้อมูลด้วยมือแก่ผู้ดูแลเว็บ (Web Master)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2
ทฤษฎี และการวิจัยที่เกี่ยวข้อง

จากเนื้อหาในบทที่ผ่านมาได้แสดงถึงโครงร่างของการวิจัยทั้งหมด ในบทที่ 2 นี้จะนำเสนอถึงทฤษฎีและการวิจัยที่เกี่ยวข้อง เพื่อเป็นพื้นฐานในการดำเนินการวิจัยต่อไป

2.1. ทฤษฎีที่เกี่ยวข้อง

2.1.1. เอชทีทีพี (HyperText Transfer Protocol: HTTP) [1] [3]

เอชทีทีพีเป็นกลไก หรือโพรโทคอล (Protocol) หลักที่ใช้ในการแลกเปลี่ยนข้อมูลกันระหว่างเว็บเซิร์ฟเวอร์ (Web Server) และไคลเอนต์ (Client) โดยถูกออกแบบมาให้มีความกระทัดรัด สามารถทำงานได้รวดเร็ว มีกระบวนการทำงานที่ไม่ซับซ้อน และมีคำสั่งที่ใช้งานไม่มากนัก แต่สามารถรองรับข้อมูลได้ทุกรูปแบบไม่ว่าจะเป็นข้อมูลทั่วไปที่เข้ารหัสแบบไมม์ (Multipurpose Internet Mail Extensions: MIME) หรือข้อมูลที่เป็นกราฟิก เช่น ไฟล์ที่เป็นจิฟ (Graphics Interchange Format: GIF) หรือเจเพ็ก (Joint Photographic Experts Group: JPEG) เป็นต้น

หลักการทำงานของเอชทีทีพีจะแบ่งออกเป็น 2 ด้านคือ ด้านเว็บเซิร์ฟเวอร์ และด้านไคลเอนต์ โดยไคลเอนต์ที่ต้องการข้อมูลจากเว็บเซิร์ฟเวอร์จะติดต่อมายังเว็บเซิร์ฟเวอร์โดยผ่านทีซีพีคอนเน็กชั่น (TCP Connection) บนพอร์ต (Port) ที่ระบุอยู่หลังยูอาร์ไอ (Uniform Resource Identifier: URI) ซึ่งค่าปกติคือพอร์ตหมายเลข 80 จากนั้นไคลเอนต์จะส่งชุดของคำร้องขอ (Request) โดยสิ้นสุดด้วยบรรทัดว่าง 1 บรรทัดเพื่อบอกให้เว็บเซิร์ฟเวอร์ทราบว่าสิ้นสุดชุดของคำร้องขอแล้ว ตัวอย่าง เช่น

```
GET  /very/important/document.html  HTTP/1.1
Host:  www.modperl.com
```

บรรทัดแรกของชุดของคำร้องขอประกอบไปด้วย 3 ส่วน ส่วนแรกคือ วิธีการร้องขอ ค่าที่เป็นไปได้คือ GET, POST, READ, PUT หรือ DELETE โดย GET หมายถึง การร้องขอเพื่อดึงข้อมูลจากเว็บเซิร์ฟเวอร์ POST หมายถึง การร้องขอเพื่อส่งข้อมูลไปให้เว็บเซิร์ฟเวอร์ทำงาน HEAD หมายถึง การร้องขอข้อมูลเกี่ยวกับเอกสารบนเว็บเซิร์ฟเวอร์โดยที่ไม่ต้องการตัวเอกสาร PUT หมายถึง การร้องขอเพื่อส่งข้อมูลใหม่ไปให้เว็บเซิร์ฟเวอร์ และ DELETE หมายถึง การร้องขอเพื่อลบข้อมูลจากเว็บเซิร์ฟเวอร์ โดยทั่วไปแล้วเว็บเซิร์ฟเวอร์จะไม่อนุญาตให้ใช้วิธีการร้องขอแบบ PUT และ DELETE ได้ ส่วนที่สองคือ ยูอาร์ไอของข้อมูลที่ต้องการ โดยอยู่ในรูปแบบของไฟล์บนระบบปฏิบัติการยูนิกซ์ (UNIX-Style) ซึ่งเว็บเซิร์ฟเวอร์จะทำการแปลงเป็นไฟล์ที่อยู่บนเว็บเซิร์ฟเวอร์อีกทีและส่วนสุดท้ายคือ โพรโทคอลที่ใช้ ในตัวอย่างเป็นโพรโทคอลเอชทีทีพี รุ่นที่ 1.1 (HTTP/1.1)

ถัดจากบรรทัดแรกจะเป็นชุดของเอชทีทีพีเฮดเดอร์ (HTTP Header) ซึ่งไคลเอนต์สามารถส่งให้กับเว็บเซิร์ฟเวอร์ได้เพื่อให้คำร้องขอทำงานได้ดียิ่งขึ้น โดยแต่ละเอชทีทีพีเฮดเดอร์ประกอบด้วย ชื่อของเอชทีทีพีเฮดเดอร์ เครื่องหมายจุดคู่ (:) และค่าของเอชทีทีพีเฮดเดอร์ ในเอชทีทีพี รุ่นที่ 1.1 เอชทีทีพีเฮดเดอร์ที่จำเป็นคือ Host ซึ่งจะระบุถึงเว็บเซิร์ฟเวอร์ที่รับคำร้องขอ ใช้เพื่อจำลองเว็บเซิร์ฟเวอร์หลาย ๆ ตัวบนเว็บเซิร์ฟเวอร์ตัวเดียว ตัวอย่างคำร้องขอจากไคลเอนต์ซึ่งใช้ Netscape Navigator เช่น

GET /news.html HTTP/1.1
Connection: Keep-Alive
User-Agent: Mozilla/4.05 [en] (X11; I; Linux 2.0.33 i686)
Host: www.modperl.com
Referer: http://www.modperl.com/index.html
If-Modified-Since: Tue, 24 Feb 1998 11:19:03 GMT
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*
Accept-Language: en
Accept-Charset: iso-8859-1, *, utf-8

สำหรับเอชทีทีพีเฮดเดอร์อื่น ๆ ได้แก่ Connection หมายถึง ต้องการให้เว็บเซิร์ฟเวอร์ยังไม่ปิดทีซีพีคอนเน็กชั่นหลังจากที่ตอบสนองต่อคำร้องขอ ทำให้เว็บเซิร์ฟเวอร์ทำงานได้อย่างมีประสิทธิภาพมากขึ้นสำหรับเว็บเพจที่มีรูปภาพในหน้าเว็บหลายรูป User-Agent หมายถึง ชนิดของโปรแกรมบราวเซอร์ (Browser) ที่ไคลเอนต์ใช้งาน Referer หมายถึง ยูอาร์ไอที่อ้างถึงข้อมูลที่ต้องการร้องขอ

If-Modified-Since หมายถึง วัน เวลา ที่ข้อมูลถูกเก็บไว้ในแคช (Cache) เว็บเซิร์ฟเวอร์จะเปรียบเทียบค่านี้กับ วัน เวลา ที่มีการแก้ไขข้อมูลครั้งหลังสุดบนเว็บเซิร์ฟเวอร์ ถ้ามีการแก้ไขข้อมูลหลังจากวัน เวลา ที่ข้อมูลถูกเก็บไว้ เว็บเซิร์ฟเวอร์จะส่งข้อมูลใหม่ไปให้ไคลเอนต์ Accept หมายถึง รายการของการเข้ารหัสแบบไมม์ที่ไคลเอนต์สามารถรองรับได้ Accept-Language หมายถึง ภาษาที่ไคลเอนต์ต้องการ และ Accept-Charset หมายถึง รหัสตัวอักษร (Character Set) ที่ไคลเอนต์สามารถแสดงผลได้ ตัวอย่างการร้องขอข้อมูลจากเว็บเซิร์ฟเวอร์โดยใช้โปรแกรม Telnet เช่น

%telnet www.modperl.com 80
Trying 207.198.250.44 …
Connected to modperl.com.
Escape character is '^]'.
GET /news.html HTTP/1.1
Host: www.modperl.com

```
HTTP/1.1   200   OK
Date:   Tue, 24 Feb 1998 13:16:02 GMT
Server:   Apache/1.3.0 (Unix) mod_perl/1.13
Last-Modified:   Wed, 11 Feb 1998 21:05:25 GMT
Etag:   “65e5a-37c-35a7d395”
Accpet-Ranges:   bytes
Content-Length:   892
Connection:   close
Content-Type:   text/html

<HTML>
<HEAD>
<TITLE>What’s New</TITLE>
</HEAD>
<BODY>
…
Connection closed by foreign host.
```

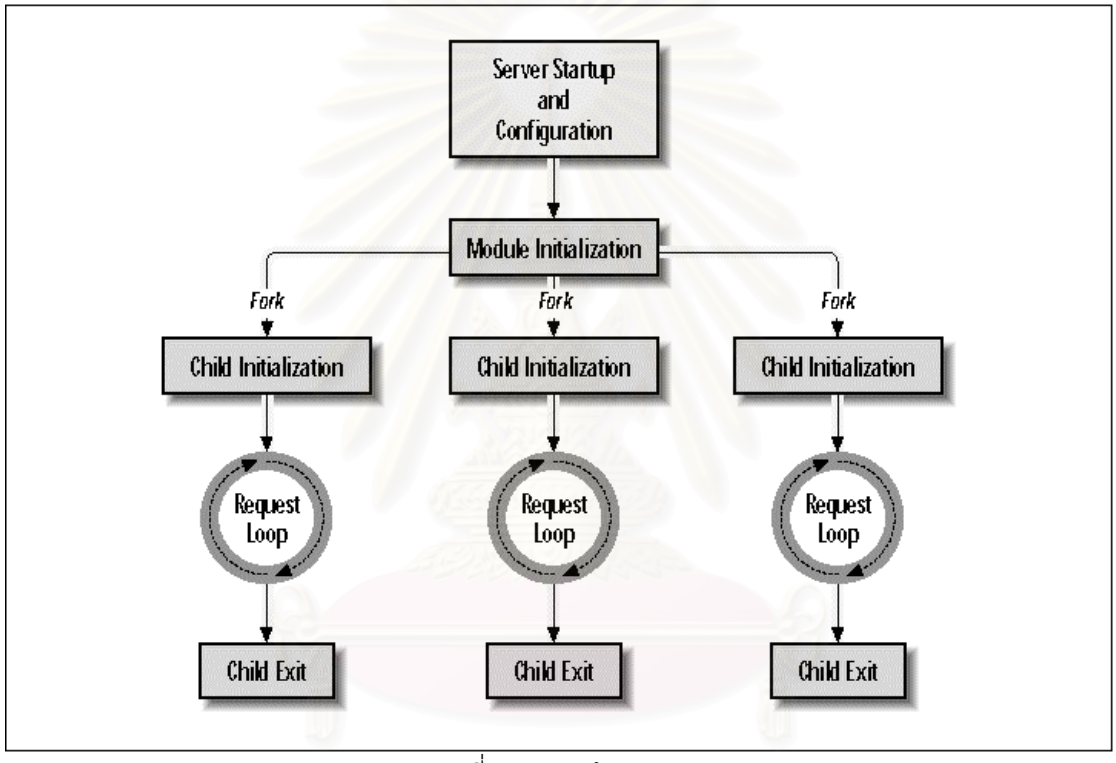
เว็บเซิร์ฟเวอร์จะตอบสนองต่อคำร้องขอโดยการส่งสถานะตามด้วยเอชทีทีพีเฮดเดอร์ และข้อมูลที่ไคลเอนต์ร้องขอโดยจะแยกจากเอชทีทีพีเฮดเดอร์ด้วยบรรทัดว่าง 1 บรรทัด

บรรทัดแรกของการตอบสนองเริ่มด้วยรุ่นของเอชทีทีพีตามด้วยรหัสแสดงสถานะ (Status Code) และข้อความอธิบายรหัสแสดงสถานะ ถัดจากบรรทัดแรกเป็นชุดของเอชทีทีพีเฮดเดอร์ โดย Date หมายถึง วัน เวลาปัจจุบัน Server หมายถึง ชื่อ และรุ่นของโปรแกรมที่ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์ Last-Modified และ Content-Length หมายถึง วัน เวลา ที่มีการแก้ไขข้อมูลครั้งหลังสุด และความยาวของข้อมูลที่จะถูกเก็บไว้ในแคช Content-Type หมายถึง การเข้ารหัสแบบไมม์ของข้อมูล

Etag หรือ Entity Tag เป็นฟิล์ดพิเศษในเอชทีทีพี รุ่นที่ 1.1 ใช้เพื่อให้การเก็บข้อมูลไว้ในแคชมีความถูกต้อง เที่ยงตรงมากขึ้น โดยจะเป็นค่าที่เป็นตัวแทนของข้อมูล ซึ่งจะมีการเปลี่ยนแปลงเมื่อมีการแก้ไขข้อมูล Accept-Range เป็นฟิล์ดพิเศษในเอชทีทีพี รุ่นที่ 1.1 เช่นกัน ใช้เพื่อให้ไคลเอนต์สามารถดึงข้อมูลจากเว็บเซิร์ฟเวอร์ต่อโดยไม่ต้องเริ่มดึงข้อมูลตั้งแต่ต้นใหม่เมื่อคอนเน็กชั่นไม่สมบูรณ์ Connection หมายถึง สถานะของคอนเน็กชั่น Content-Type หมายถึง การเข้ารหัสแบบไมม์ของข้อมูล หลังจากชุดของเอชทีทีพีเฮดเดอร์แล้วจะตามด้วยข้อมูลที่ไคลเอนต์ทำการร้องขอ โดยข้อมูลต้องมีความยาวเท่ากับค่าที่ระบุใน Content-Length และเข้ารหัสแบบไมม์ตามที่ระบุใน Content-Type ด้วย

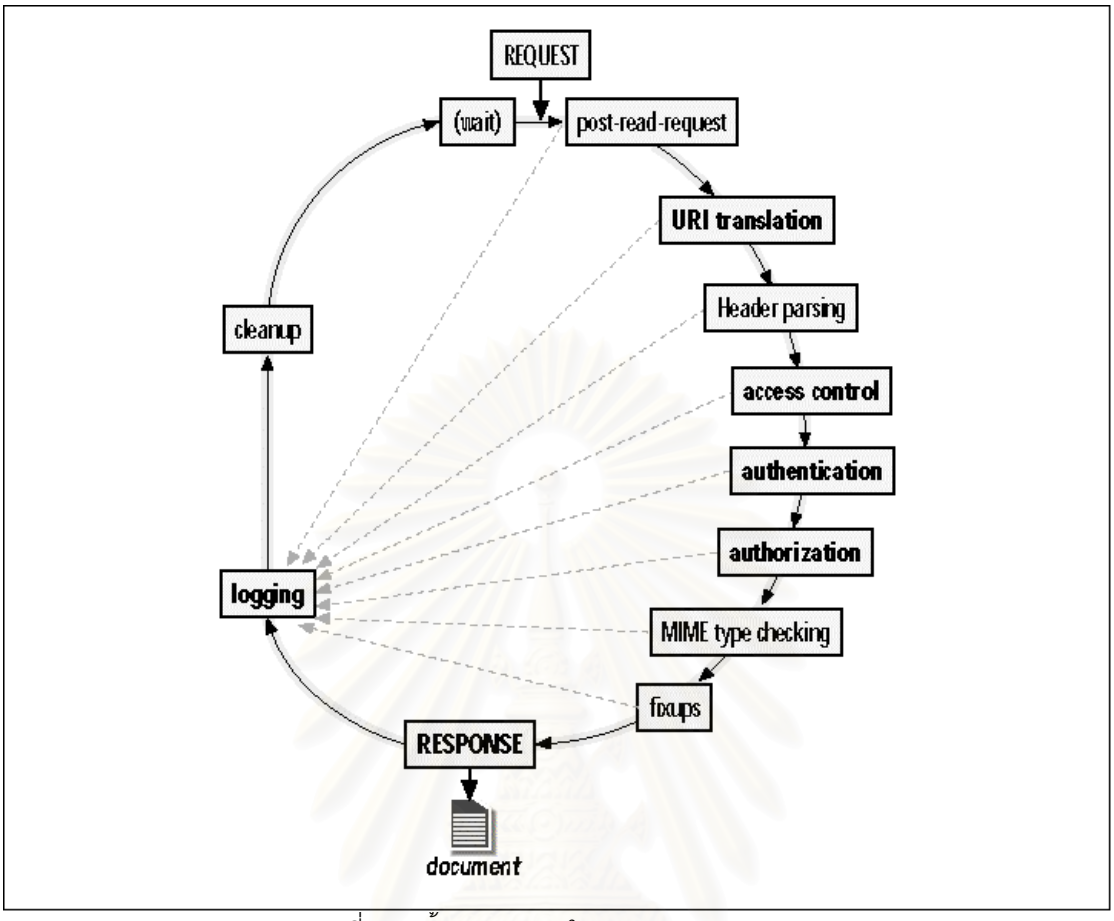
2.1.2. อาปาเช่ (Apache) [2] [3]

การทำงานของอาปาเช่แสดงดังรูปที่ 2.1 โดยหลังจากเริ่มต้นทำงาน (Startup) แล้วจะทำการกำหนดค่าเริ่มต้นต่าง ๆ (Initialization) และสำเนาตัวเองออกเป็นหลาย ๆ โปรเซส (Fork) หลังจากนั้นจะเข้าสู่กระบวนการวนลูป (Request Loop) เพื่อรอรับคำร้องขอจากไคลเอนต์ เมื่อตอบสนองต่อคำร้องขอเสร็จแล้วก็จะออกจากการวนลูปและหยุดการทำงานลง (Exit) เหตุการณ์ที่น่าสนใจต่าง ๆ จะเกิดขึ้นในระหว่างการวนลูป ผู้พัฒนาสามารถแทรกการทำงานต่าง ๆ ที่ต้องการในระหว่างการวนลูปได้โดยใช้โมดูล (Module) ซึ่งเขียนจากภาษาซี (C Language) หรือเพิร์ล (Perl) รูปที่ 2.2 แสดงขั้นตอนต่าง ๆ ในระหว่างการวนลูป



รูปที่ 2.1 การทำงานของอาปาเช่

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 2.2 ขั้นตอนต่าง ๆ ในระหว่างการวนลูปของอาปาเช่

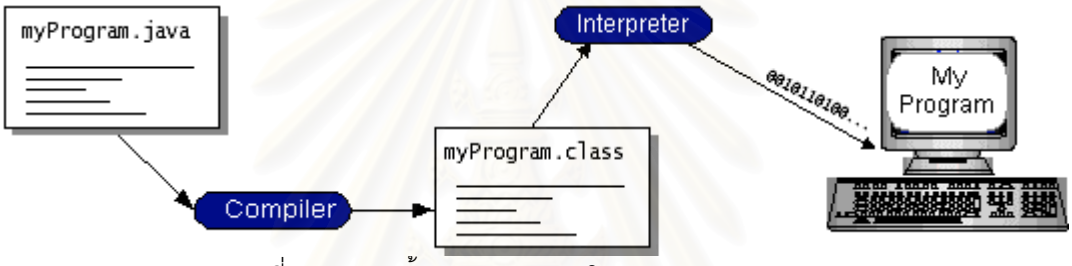
ระหว่างขั้นตอนต่าง ๆ ในการวนลูปอาปาเช่จำเป็นต้องทำการตัดสินใจหลาย ๆ อย่างเพื่อจัดการ (Handler) กับคำร้องขอต่าง ๆ ผู้พัฒนาสามารถพัฒนาโมดูลขึ้นเพื่อช่วยจัดการคำร้องขอต่าง ๆ ได้ตามที่ต้องการ โดยอาปาเช่จะจัดการแบบปกติกับคำร้องขอใด ๆ ที่ไม่มีโมดูลรองรับ ด้านล่างแสดงการตัดสินใจต่าง ๆ ในระหว่างการวนลูป

1. ข้อมูลที่ทำการร้องขอ (URI Translate Phase)
2. ที่มาของคำร้องขอ (Access Control Phase)
3. เจ้าของคำร้องขอ (Authentication Phase)
4. สิทธิบนข้อมูลที่ทำการร้องขอ (Authorization Phase)
5. การเข้ารหัสแบบไมม์ของข้อมูลที่ทำการร้องขอ (MIME Type Checking Phase)
6. การสร้างข้อมูลเพื่อตอบสนองต่อคำร้องขอ (Response Phase)
7. การเก็บล็อคของคำร้องขอ (Logging Phase)
8. การจัดการทรัพยากรหลังจากเสร็จสิ้นการตอบสนองต่อคำร้องขอ (Cleanup Phase)

2.1.3. แอปเพล็ต (Applet) [5] [6]

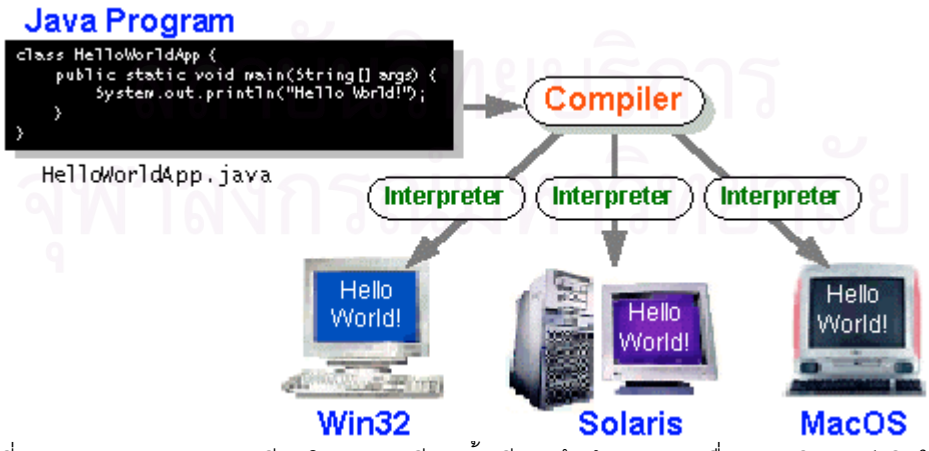
จาวาเป็นเทคโนโลยีที่ถูกพัฒนาขึ้นโดยบริษัทซันไมโครซิสต็ม (Sun Microsystems) โดยมีเป้าหมายคือ การพัฒนาโปรแกรมที่มีลักษณะไม่ขึ้นกับแพลตฟอร์ม (Platform – Independent) หมายถึง โปรแกรมที่เขียนขึ้นสามารถทำงานได้บนฮาร์ดแวร์ และซอฟต์แวร์ใด ๆ โดยไม่จำเป็นต้องมีการแก้ไขโปรแกรม องค์ประกอบของจาวาสามารถแบ่งออกได้เป็น 2 ส่วน ได้แก่

1. ภาษาจาวา เป็นภาษาโปรแกรม (Programming Language) ที่มีการทำงานทั้งในลักษณะคอมไพล์ (Compile) และอินเทอร์พรีต (Interpret) โดยคอมไพเลอร์ (Compiler) จะทำการแปลงโปรแกรมภาษาจาวาไปเป็น Intermediate Language ที่มีลักษณะไม่ขึ้นกับแพลตฟอร์มเรียกว่า จาวาไบต์โค้ด (Java Bytecode) และอินเทอร์พรีเตอร์ (Interpreter) จะทำการแปลจาวาไบต์โค้ดให้ทำงานอีกครั้งหนึ่ง ขั้นตอนการพัฒนาโปรแกรมภาษาจาวาแสดงดังรูปที่ 2.3



รูปที่ 2.3 แสดงขั้นตอนการพัฒนาโปรแกรมภาษาจาวา

จาวาไบต์โค้ดนี้เปรียบเสมือนภาษาเครื่อง (Machine Code) ของ Java Virtual Machine (Java VM) จาวาอินเทอร์พรีเตอร์ต่าง ๆ ถูกพัฒนาขึ้นตามข้อกำหนดของ Java VM นี้ ด้วยการใช้จาวาไบต์โค้ดทำให้สามารถเขียนโปรแกรมเพียงครั้งเดียวแล้วนำไปทำงานบนเครื่องคอมพิวเตอร์ชนิดใด ๆ ก็ได้ (Write Once, Run Anywhere) แสดงดังรูปที่ 2.4

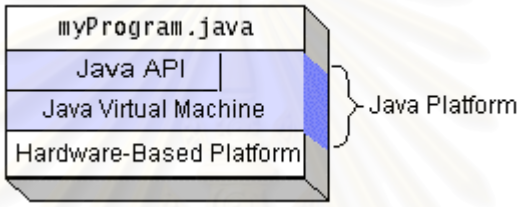


รูปที่ 2.4 แสดงลักษณะการเขียนโปรแกรมเพียงครั้งเดียวแล้วทำงานบนเครื่องคอมพิวเตอร์ชนิดใด ๆ

2. จาวาแพลตฟอร์ม (Java Platform) คือ ซอฟต์แวร์แพลตฟอร์มที่ทำงานอยู่บนฮาร์ดแวร์แพลตฟอร์มต่าง ๆ โดยประกอบด้วยองค์ประกอบ 2 ส่วน ได้แก่

   - Java Virtual Machine (Java VM)
   - Java Application Programming Interface (Java API)

   Java VM เป็นพื้นฐานของจาวาแพลตฟอร์ม และถูกพัฒนาสำหรับฮาร์ดแวร์แพลตฟอร์มต่าง ๆ Java API คือ API มาตรฐานที่ช่วยอำนวยความสะดวกในการเขียนโปรแกรมให้กับผู้พัฒนา เช่น กราฟิกสำหรับติดต่อกับผู้ใช้ (Graphic User Interface: GUI) เป็นต้น โดยจะแบ่ง API ออกเป็นกลุ่ม ๆ เรียกว่า แพกเก็จ (Package) รูปที่ 2.5 แสดงองค์ประกอบของจาวาแพลตฟอร์ม



รูปที่ 2.5 แสดงองค์ประกอบของจาวาแพลตฟอร์ม

Java Applet เป็นโปรแกรมภาษา Java ที่ทำงานร่วมกับเอชทีเอ็มแอลเพจ (HTML Page) โดยจะถูกส่งไปกับเอชทีเอ็มแอลเพจให้ไปทำงานภายใต้โปรแกรมบราวเซอร์ที่มีจาวาอินเทอร์พรีเตอร์ บนเครื่องที่รับเอชทีเอ็มแอลเพจนั้นไป จาวาแอปเพล็ตไม่สามารถทำงานได้ด้วยตัวเองต้องทำงานในสภาวะแวดล้อมที่กำหนดโดยบราวเซอร์ และถูกบราวเซอร์ควบคุมการดำเนินของโปรแกรมให้ผ่านขั้นตอนต่าง ๆ

2.1.4. การบีบอัดคงสัญญาณหลัก (Lossy Compression)

การบีบอัด (Compression) สามารถแบ่งออกได้เป็น 2 ประเภทคือ การบีบอัดคงสัญญาณ (Lossless Compression) และการบีบอัดคงสัญญาณหลัก (Lossy Compression) โดยการบีบอัดแบบคงสัญญาณ ข้อมูลในต้นฉบับจะไม่เกิดการสูญหายในระหว่างทำการบีบอัด ดังนั้นจะสามารถสร้างข้อมูลต้นฉบับจากข้อมูลที่ถูกบีบอัดแล้วได้ แต่การบีบอัดแบบนี้จะใช้เวลามาก และสามารถลดขนาดของข้อมูลได้น้อย ส่วนการบีบอัดแบบคงสัญญาณหลักนั้นข้อมูลในต้นฉบับบางส่วนจะเกิดการสูญหายในระหว่างทำการบีบอัด ทำให้ไม่สามารถสร้างข้อมูลต้นฉบับจากข้อมูลที่ถูกบีบอัดแล้วได้ แต่จะใช้เวลาน้อย และสามารถลดขนาดของข้อมูลได้มากกว่าการบีบอัดแบบคงสัญญาณ

ในการแสดงตัวอย่างข้อมูลภาพ และภาพเคลื่อนไหวจะใช้การบีบอัดแบบคงสัญญาณหลัก ทั้งนี้เนื่องจากต้องการข้อมูลที่มีขนาดเล็ก วิธีการบีบอัดที่รวดเร็ว และไม่ต้องการความถูกต้องของข้อมูลมากนัก

2.1.5. ข้อมูลมัลติมีเดียในอินเทอร์เน็ต

2.1.5.1. จิฟ (Graphic Interchange Format: GIF)

ไฟล์ข้อมูลภาพประเภทนี้เป็นการบีบอัดแบบคงสัญญาณหลัก จำนวนสีไม่เกิน 256 สี (8 บิต) มีขนาดเล็ก เหมาะสำหรับข้อมูลภาพที่มีความแตกต่างของสีชัดเจน เช่น ข้อมูลภาพการ์ตูน โลโก้ ไอคอน ข้อความ เส้น หรือปุ่มต่าง ๆ ไฟล์จิฟมี 2 รูปแบบคือ GIF87 และ GIF89a โดยไฟล์จิฟที่ใช้กันอยู่คือ GIF89a ซึ่งมีคุณสมบัติเพิ่มขึ้นจาก GIF87 ดังนี้

Interlaced

คุณสมบัตินี้จะทำให้ข้อมูลภาพที่ปรากฏบนเว็บเพจค่อย ๆ ปรากฏจากข้อมูลภาพที่มีความละเอียดต่ำจนกระทั่งได้ข้อมูลภาพที่มีความละเอียดปกติซึ่งจะช่วยให้ผู้ใช้งานรู้สึกว่าเวลาที่ต้องรอสั้นลง แต่ข้อเสียของคุณสมบัตินี้คือ ทำให้ไฟล์ข้อมูลภาพที่ได้มีขนาดใหญ่ขึ้น

Transparency

คุณสมบัตินี้จะช่วยทำให้พื้นหลังของข้อมูลภาพไม่ปรากฏจึงมีประโยชน์มากในการแสดงข้อมูลภาพบนเว็บเพจที่พื้นหลังมีลักษณะเป็นลวดลาย หรือใช้กับข้อมูลภาพที่มีรูปร่างไม่ใช่เรขาคณิตเพื่อทำให้ผู้ใช้งานมองไม่เห็นขอบเขตของข้อมูลภาพ ส่งผลให้การนำข้อมูลภาพไปตกแต่งเว็บเพจมีความสวยงามมากขึ้น

GIF Animation

เป็นข้อมูลภาพที่สามารถเคลื่อนไหวได้โดยใช้วิธีเก็บข้อมูลภาพจำนวนหนึ่งซึ่งเป็นข้อมูลภาพที่มีการกระทำต่อเนื่องกันเป็นขั้นตอนไว้ในไฟล์เดียว (ใน 1 ไฟล์มีหลายเฟรม) โดยมีการกำหนดเวลาในการหมุนเวียนกันขึ้นมาแสดงและแสดงในลักษณะวนซ้ำ

2.1.5.2. เจเพ็ก (Joint Photographic Expert Group: JPEG, JPG)

ไฟล์ข้อมูลภาพประเภทนี้สามารถแสดงสีได้ถึง 16.7 ล้านสี (24 บิต) โดยมีขนาดไฟล์ไม่ใหญ่มากนัก เนื่องจากเป็นการบีบอัดแบบคงสัญญาณหลัก ซึ่งสามารถกำหนดระดับคุณภาพของไฟล์ได้ โดยระดับคุณภาพที่น้อยจะทำให้ไฟล์ที่ได้มีขนาดเล็กลง แต่การใช้ระดับคุณภาพที่น้อยเกินไปจะส่งผลให้ไฟล์ที่ได้ไม่สามารถนำไปใช้งานได้ ไฟล์เจเพ็กนี้เหมาะสำหรับข้อมูลภาพที่ละเอียด มีการใช้สีจำนวนมาก เช่น ข้อมูลภาพถ่าย หรือ Grayscale

คุณสมบัติที่น่าสนใจของไฟล์เจเพ็ก คือ คุณสมบัติ Progressive ซึ่งมีลักษณะการทำงานเหมือนคุณสมบัติ Interlacing ของไฟล์จิฟ

2.1.5.3. ปิง (Portable Network Graphics: PNG)

สำหรับไฟล์ข้อมูลภาพประเภทนี้ยังไม่ค่อยมีการใช้งานบนอินเทอร์เน็ตมากนัก อาจเนื่องจากเป็นไฟล์ที่เกิดขึ้นมาภายหลัง จุดประสงค์ในการสร้างไฟล์ปิงคือ เพื่อนำมาใช้งานทดแทนไฟล์จิฟ ซึ่งมีผู้พัฒนาโปรแกรมกราฟิกจะต้องเสียค่าลิขสิทธิ์เพื่อให้มีสิทธิใช้ไฟล์จิฟได้ในโปรแกรมที่พัฒนาขึ้น

ไฟล์ปิงมีคุณสมบัติ Interlacing และ Transparency เช่นเดียวกับไฟล์จิฟ และยังได้รับการพัฒนาความสามารถอื่น ๆ เพิ่มเติมอีกด้วย เช่น สามารถแสดงสีได้มากถึง 16.7 ล้านสี (24 บิต) เช่นเดียวกับไฟล์เจเพ็ก แต่ไฟล์ปิงนี้ไม่สนับสนุนคุณสมบัติแอนนิเมชั่น (Animation)

2.1.5.4. เอ็มเพ็ก (Moving Picture Experts Group: MPEG, MPG)

เป็นวิธีการบีบอัดแบบคงสัญญาณหลักของไฟล์ข้อมูลประเภทดิจิตอลวีดีโอ (Digital Video) ถูกพัฒนาขึ้นโดย Moving Picture Experts Group เอ็มเพ็กสามารถบีบอัดข้อมูลได้สูงเนื่องจากจะเก็บข้อมูลเฉพาะส่วนที่มีการเปลี่ยนแปลงเท่านั้น โดยจะใช้เทคนิคที่เรียกว่า DCT (Discrete Cosine Transform) ในการบีบอัดข้อมูล ปัจจุบันมีมาตรฐานหลักออกมา 3 มาตรฐานได้แก่ เอ็มเพ็ก-1 (MPEG-1) เอ็มเพ็ก-2 (MPEG-2) และเอ็มเพ็ก-4 (MPEG-4)

เอ็มเพ็ก-1 สามารถแสดงข้อมูลวีดีโอ (Video) ด้วยความละเอียด 352 x 240 จุด ที่ความเร็ว 30 เฟรมต่อวินาที (fps) ถูกนำไปใช้สำหรับวีดีโอซีดี (Video-CD) และซีดีไอ (Compact Disc – interactive: CD-I)

เอ็มเพ็ก-2 สามารถแสดงข้อมูลวีดีโอ ด้วยความละเอียด 720 x 480 และ 1280 x 720 จุด ที่ความเร็ว 60 เฟรมต่อวินาที ด้วยคุณภาพเสียงระดับซีดี (Compact Disc: CD) เอ็มเพ็ก-2 ถูกนำไปใช้สำหรับดิจิตอลทีวี บรอดเคส (Digital TV Broadcast) และดีวีดี (DVD)

เอ็มเพ็ก-4 เป็นวิธีการบีบอัดข้อมูลกราฟิก และวีดีโอที่มีพื้นฐานมาจากเอ็มเพ็ก-1 และเอ็มเพ็ก-2 ถูกนำไปใช้ในการส่งข้อมูลภาพ และวีดีโอบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณแคบ นอกจากนี้ยังสามารถผสมข้อมูลวีดีโอเข้ากับข้อความ กราฟิก 2D และ 3D แอนนิเมชั่นได้อีกด้วย

2.1.5.5. Video for Windows

ถูกพัฒนาขึ้นโดยบริษัทไมโครซอฟต์สำหรับเก็บข้อมูลวีดีโอและ ออดิโอ (Audio) มีนามสกุลเป็น .AVI โดยถูกจำกัดความละเอียดที่ 320 x 240 จุด ที่ความเร็ว 30 เฟรมต่อวินาที ไฟล์ข้อมูลประเภทนี้ไม่ต้องการฮาร์ดแวร์โดยเฉพาะในการทำงาน

2.1.5.6. ควิกไทม์ (QuickTime)

เป็นไฟล์สำหรับข้อมูลวีดีโอและ แอนนิเมชั่น พัฒนาโดยบริษัทแอปเปิล ถูกรวมเข้ากับระบบปฏิบัติการของเครื่อง Macintosh ทำให้มีการใช้งานกันอย่างกว้างขวาง เครื่องคอมพิวเตอร์ส่วนบุคคลที่ต้องการใช้ไฟล์ QuickTime ต้องการ QuickTime Driver สำหรับทำงาน มีนามสกุลเป็น .QT, .MOV หรือ .MOOV

2.2. การวิจัยที่เกี่ยวข้อง

2.2.1. การวิจัยเรื่อง Bringing Treasures to the Surface: Previews and Overviews in a Prototype for the Library of Congress National Digital Library [14] [15]

Catherine Plaisant, Gary Marchionini และ Anita Komlodi จาก Human-Computer Interaction Laboratory and Digital Library Research Group, University of Maryland ได้ทำการออกแบบ และพัฒนา Interface Design Component สำหรับ American Memory Collection of Historical Multimedia Materials ของ Library of Congress (LC) โดยได้แสดงให้เห็นถึงประโยชน์ในการแสดงตัวอย่าง เช่น Thumbnail Video ตัวอย่าง และภาพรวมของข้อมูล เช่น สารบัญของข้อมูลต่าง ๆ ซึ่งทำให้ผู้ใช้สามารถค้นหาและเข้าถึงข้อมูลที่ต้องการได้สะดวก รวดเร็วขึ้น

2.2.2. การวิจัยเรื่อง Soft Caching: Web Cache Management Techniques for Images [16] และ Design and Implementation of a Soft Caching Proxy [17]

Antonio Ortega, Fabio Carignano, Serge Ayer และ Martin Vetterli ได้นำเสนอเทคนิค Soft Caching สำหรับจัดการข้อมูลภาพใน Web Cache เพื่อเพิ่มประสิทธิภาพการทำงานของ Proxy Server โดยจะทำการลดคุณภาพของข้อมูลภาพที่ถูกเก็บใน Web Cache เพื่อลดเวลาที่ใช้ในการรับส่งข้อมูลระหว่างเว็บเซิร์ฟเวอร์ และไคลเอนต์ และ Jussi Kangasharju, Young Gap Kwon และ Antonio Ortega จาก Integrated Media Systems Center, Department of EE-Systems, University of Southern California ได้ทำการออกแบบ และพัฒนา Soft Caching Proxy ขึ้นโดยใช้ Squid Proxy เป็นพื้นฐานในการพัฒนา

2.2.3. การวิจัยเรื่อง Web Server Workload Characterization [18]

Martin F. Arlitt และ Carey L. Williamson จาก Department of Computer Science, University of Saskatchewan ได้ทำการศึกษา และวิเคราะห์ลักษณะข้อมูลในอินเทอร์เน็ตเว็บเซิร์ฟเวอร์ โดยเก็บข้อมูลจากเว็บเซิร์ฟเวอร์ในสภาพแวดล้อมต่าง ๆ ได้แก่ สถาบันการศึกษา สถาบันวิจัย และผู้ให้บริการอินเทอร์เน็ต มีวัตถุประสงค์เพื่อหาลักษณะข้อมูลในอินเทอร์เน็ตเพื่อเป็นข้อมูลในการปรับปรุงประสิทธิภาพในการทำงานของ World Wide Web

2.2.4. การวิจัยเรื่อง An Investigation of Documents from the World Wide Web [19]

Allison Woodruff, Paul M. Aoki, Eric Brewer, Paul Gauthier และ Lawerence A. Rowe จาก Computer Science Division, University of California at Berkeley ได้ทำการวิเคราะห์เว็บเพจในอินเทอร์เน็ต โดยใช้ข้อมูลที่รวบรวมได้จาก Inktomi Web Crawler ข้อมูลที่นำมาวิเคราะห์ ได้แก่ ขนาดของเอกสาร อัตราส่วนระหว่างขนาดของเอกสารกับจำนวนของ Tag การใช้งานของ Tag และ Attribute โพรโทคอล พอร์ต ประเภทของไฟล์ จำนวน Link ที่ปรากฏในเว็บเพจ ความซับซ้อนของเอกสาร และลักษณะต่าง ๆ ที่ผิดไวยากรณ์ (Syntax Error)

2.2.5. การวิจัยเรื่องการวิเคราะห์ความสัมพันธ์ระหว่างลักษณะของยูอาร์แอลกับการใช้งานเว็บแคช [20]

คุณนพรัตน์ พรมมหาไชย ได้ทำการวิเคราะห์ความสัมพันธ์ระหว่างลักษณะของยูอาร์แอลกับการใช้งานเว็บแคชในการใช้อินเทอร์เน็ต โดยใช้ข้อมูลของการบริการเว็บแคชของจุฬาลงกรณ์มหาวิทยาลัย ข้อมูลที่ทำการวิเคราะห์ ได้แก่ ความยาว ความลึกของเส้นทาง และนามสกุลแฟ้มข้อมูลของยูอาร์แอล ตัววัดที่ใช้ คือ อัตราส่วนของจำนวนข้อมูลที่พบในแคชเทียบกับจำนวนครั้งที่ให้บริการทั้งหมด และอัตราส่วนของปริมาณข้อมูลที่พบในแคชเทียบกับปริมาณข้อมูลให้บริการทั้งหมด

สำหรับซอฟต์แวร์ที่ใช้ในการแสดงตัวอย่างข้อมูลภาพ และภาพเคลื่อนไหว ผู้วิจัยได้ค้นหา (Search) จาก http://sourceforge.net/ และ http://freshmeat.net/ ซึ่งเป็นเว็บไซต์สำหรับการพัฒนาซอฟต์แวร์ โอเพน ซอร์ส (Open Source Software) และ http://www.google.com/ ซึ่งเป็นบริการเสิร์ชเอ็นจิ้น (Search Engine) ที่ได้รับความนิยมสูง โดยซอฟต์แวร์ที่ต้องการจะต้องมีลักษณะเป็น Command Line หรือเป็น Application Programming Interface (API) ที่ทำงานได้บนยูนิกซ์แพลตฟอร์ม

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3
การออกแบบ และพัฒนาระบบ

เนื้อหาในบทที่ 3 นี้จะเป็นการออกแบบ และพัฒนาระบบแสดงตัวอย่างข้อมูลจากเว็บเพจสำหรับการเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำ ซึ่งแบ่งออกเป็นหัวข้อต่าง ๆ ได้ดังนี้

1. การตรวจสอบความกว้างช่องสัญญาณ
2. การเก็บข้อมูลความกว้างช่องสัญญาณ
3. ค่าความล่าช้า
4. การแสดงตัวอย่างของข้อมูล
5. อินเตอร์เฟส
6. เครื่องมือที่ใช้ในการวิจัย

3.1. การตรวจสอบความกว้างช่องสัญญาณ

3.1.1. การตรวจสอบความกว้างช่องสัญญาณโดยการ Ping

การตรวจสอบความกว้างช่องสัญญาณจะวัดได้จากระยะเวลาที่แพกเก็จเดินทางไปกลับระหว่างเว็บเซิร์ฟเวอร์และไคลเอนต์ โดยใช้ Net::Ping และ Net::Ping::External Perl Module [11] [12] ในการทำงาน

Net::Ping และ Net::Ping::External Perl Module เป็น Perl Module ที่ใช้ทดสอบว่าเครื่องปลายทางที่ต้องการติดต่อในเน็ตเวิร์คสามารถติดต่อได้หรือไม่ (Reachability) โดยการทำงานจะส่งแพกเก็จจากเครื่องต้นทางไปยังเครื่องปลายทาง และรอแพกเก็จตอบรับจากเครื่องปลายทางกลับมา สามารถเลือกใช้โพรโทคอลในการส่งแพ็กเก็จได้ 5 โพรโทคอล ได้แก่ TCP, UDP, ICMP, External และ SYN ถ้าไม่ได้ระบุโพรโทคอลที่ต้องการใช้ โมดูลจะใช้โพรโทคอล TCP ในการติดต่อ รายละเอียดของโพรโทคอลต่าง ๆ แสดงดังตารางที่ 3.1

ตารางที่ 3.1 แสดงรายละเอียดของโพรโทคอลต่าง ๆ
ที่ใช้งานโดย Net::Ping และ Net::Ping::External Perl Module

โพรโทคอล	การทำงาน	สิทธิในการใช้งาน	หมายเหตุ
TCP	ติดต่อไปยัง ECHO พอร์ตของเครื่องปลายทาง	ไม่ต้องการ	Overhead สูง
UDP	ติดต่อไปยัง ECHO พอร์ตของเครื่องปลายทาง	ไม่ต้องการ	
ICMP	ส่ง ICMP ECHO Message ไปยังเครื่องปลายทาง	ต้องการสิทธิ Root	
External	ใช้ Net::Ping::External Perl Module ในการ Ping ไปยังเครื่องปลายทาง โดย Net::Ping::External Perl Module จะติอต่อกับโปรแกรม Ping ของระบบในการทำงาน	ไม่ต้องการ	
SYN	ส่ง TCP SYN ไปยังเครื่องปลายทาง	ไม่ต้องการ	

จากตารางที่ 3.1 โพรโทคอล TCP จะทำให้เกิด Overhead มากกว่า UDP หรือ ICMP ประมาณ 2.5 เท่าหรือมากกว่า และโพรโทคอล ICMP เท่านั้นที่ต้องการสิทธิ Root ในการทำงาน ถึงแม้ว่าโพรโทคอล TCP และ UDP ไม่ต้องใช้สิทธิ Root ในการทำงานแต่อุปกรณ์เน็ตเวิร์คบางอุปกรณ์ที่ไม่ได้เปิดใช้งาน ECHO พอร์ตไว้ก็จะไม่สามารถตรวจสอบได้ สำหรับเน็ตเวิร์คที่มีการจราจร (Traffic) หนานแน่นแพกเก็จจะใช้เวลาในการเดินทางมากกว่าปกติ ดังนั้นหากตั้งค่า Timeout ไว้น้อยเกินไปจะทำให้โมดูลรายงานว่าไม่สามารถติดต่อเครื่องปลายทางได้

ในการวิจัยนี้เลือกใช้โพรโทคอล External ในการตรวจสอบความกว้างช่องสัญญาณ เนื่องจากไคลเอนต์อาจไม่ได้เปิดใช้งาน TCP หรือ UDP ECHO พอร์ตไว้ซึ่งจะทำให้เว็บเซิร์ฟเวอร์ไม่สามารถตรวจสอบความกว้างช่องสัญญาณที่ติดต่อได้ และไม่สามารถใช้โพรโทคอล ICMP ได้ เนื่องจากเว็บเซิร์ฟเวอร์ไม่ได้ใช้สิทธิ Root ในการตอบสนองคำร้องขอจากไคลเอนต์

3.1.2. การตรวจสอบความกว้างช่องสัญญาณโดยการใช้แอปเพล็ต

เนื่องจากการตรวจสอบความกว้างช่องสัญญาณโดยการ Ping ไม่สามารถใช้งานได้กับไคลเอนต์ที่มีการติดตั้งไฟล์วอล์ (Firewall) ที่ไม่อนุญาตให้โพรโทคอล ICMP ผ่านเข้าออกได้ ดังนั้นวิธีการตรวจสอบความกว้างช่องสัญญาณอีกวิธีหนึ่งคือ การใช้แอปเพล็ตอ่านไฟล์ข้อมูลจากเว็บเซิร์ฟเวอร์ แล้วทำการจับเวลาที่ใช้ ในการวิจัยนี้เลือกใช้ไฟล์ข้อมูลขนาด 2 กิโลไบต์ (Kilobyte: KB) ในการทดสอบ โดยความกว้างช่องสัญญาณสามารถคำนวณได้จากสูตร

$$\text{ความกว้างช่องสัญญาณ} = \frac{\text{ขนาดของไฟล์ข้อมูล}}{\text{ระยะเวลาที่ใช้ในการอ่านไฟล์ข้อมูล}}$$

ด้วยวิธีการนี้ทำให้สามารถตรวจสอบความกว้างช่องสัญญาณของไคลเอนต์ที่มีการติดตั้งไฟล์วอล์ที่ไม่อนุญาตให้โพรโทคอล ICMP ผ่านเข้าออกได้ นอกจากนี้เมื่อไคลเอนต์ได้รับแอปเพล็ตจากเว็บเซิร์ฟเวอร์แล้ว การตรวจสอบความกว้างช่องสัญญาณครั้งต่อไปสามารถทำงานได้ทันทีโดยไม่ต้องรอรับแอปเพล็ตจากเว็บเซิร์ฟเวอร์อีกครั้งเนื่องจากแอปเพล็ตยังคงอยู่ในหน่วยความจำของไคลเอนต์ทำให้ช่วยลดเวลาในการทำงานลงได้ แต่การใช้แอปเพล็ตก็จะเป็นการเพิ่มขั้นตอนการทำงานให้แก่ระบบ และโปรแกรมบราวเซอร์ที่ไคลเอนต์ใช้จะต้องมีจาวาอินเทอร์พรีเตอร์ติดตั้งอยู่ด้วย (Java Technology – Enabled Browser) จึงจะสามารถทำงานได้

3.2. การเก็บข้อมูลความกว้างช่องสัญญาณ

เนื่องจากสถาปัตยกรรมเอชทีทีพี (HTTP Architecture) เป็นแบบ Stateless กล่าวคือ เมื่อสิ้นสุดแต่ละ HTTP Transaction แล้ว เว็บเซิร์ฟเวอร์จะไม่เก็บข้อมูลของ Transaction นั้น ๆ ไว้เลย ทำให้แม้ว่ามี Transaction จากไคลเอนต์เครื่องเดิมเข้ามาอีก เว็บเซิร์ฟเวอร์จะถือว่าเป็น Transaction ใหม่ทั้งหมด

วิธีการเก็บข้อมูลของ HTTP Transaction หลัก ๆ สามารถแบ่งออกได้เป็น 6 วิธี ดังนี้

1. จัดเก็บใน Hidden Fields
2. จัดเก็บใน Cookies
3. จัดเก็บใน URI

4. จัดเก็บใน Web Server Process Memory
5. จัดเก็บใน File
6. จัดเก็บใน Database

จากรายการข้างต้นสามารถแบ่งออกได้เป็น 2 กลุ่ม คือ วิธีที่ 1 – 3 เป็นการเก็บข้อมูลที่ฝั่งไคลเอนต์ และวิธีที่ 4 – 6 เป็นการเก็บข้อมูลที่ฝั่งเว็บเซิร์ฟเวอร์ ข้อดีสำหรับการเก็บข้อมูลที่ฝั่งไคลเอนต์ คือ ใช้ Overhead ของเว็บเซิร์ฟเวอร์น้อย เนื่องจากไม่ต้องใช้หน่วยความจำในการเก็บข้อมูล ไม่ต้องทำการค้นหาข้อมูลจากฐานข้อมูล ไม่มีการคำนวณที่ซับซ้อน แต่มีข้อเสีย คือ ต้องอาศัยการทำงานร่วมกับโปรแกรมบราวเซอร์ที่ฝั่งไคลเอนต์ การจัดเก็บข้อมูลใน Hidden Fields ใน HTML Form ทำให้ผู้ใช้สามารถดูข้อมูลที่จัดเก็บไว้ได้ นอกจากนี้ยังสามารถแก้ไขข้อมูลเหล่านี้ได้อีกด้วย การจัดเก็บข้อมูลใน Cookies จะมีปัญหากับโปรแกรมบราวเซอร์ของไคลเอนต์ที่ไม่สนับสนุน หรือไคลเอนต์ที่ปิดการทำงานของ Cookies ด้วยเหตุผลด้านความเป็นส่วนตัวอีกด้วย นอกจากนี้สำหรับข้อมูลที่มีปริมาณมากการเก็บข้อมูลที่ฝั่งไคลเอนต์จะเป็นการใช้งานความกว้างช่องสัญญาณสูงอีกด้วย

การเก็บข้อมูลที่ฝั่งเว็บเซิร์ฟเวอร์จะเป็นการแก้ปัญหาของฝั่งไคลเอนต์ที่กล่าวมาข้างต้น แต่จะนำไปสู่ปัญหาใหม่เช่นกัน การเก็บข้อมูลที่ฝั่งเว็บเซิร์ฟเวอร์จะเป็นการสร้าง Session Object ขึ้นบนเว็บเซิร์ฟเวอร์ โดยจะเก็บข้อมูลที่เกี่ยวข้องกับ User Session นั้น ๆ ถ้า User Session มีการใช้งานข้อมูลไม่มากสามารถจัดเก็บข้อมูลในหน่วยความจำของเว็บเซิร์ฟเวอร์ได้ แต่ถ้ามีการใช้งานข้อมูลปริมาณมาก ใช้งานเป็นเวลานาน จะใช้การจัดเก็บข้อมูลบน File หรือ Database แทน เนื่องจากข้อมูลที่เก็บอยู่ที่ฝั่งเว็บเซิร์ฟเวอร์ทำให้ไม่มีปัญหาเกี่ยวกับการดูข้อมูล หรือการเปลี่ยนแปลงข้อมูลจากไคลเอนต์ อย่างไรก็ตาม การเก็บข้อมูลที่ฝั่งเว็บเซิร์ฟเวอร์มีความซับซ้อนมากกว่า เนื่องจากจะต้องจัดการกับข้อมูลของหลาย ๆ Session พร้อม ๆ กัน สิ่งที่จะต้องคำนึงถึง เช่น การเข้าใช้ข้อมูลพร้อมกันของแต่ละ Session การลบ Session Object ที่ไม่ได้ใช้งาน การอ้างอิงระหว่าง Session Object และไคลเอนต์ ระยะเวลาที่สามารถเก็บข้อมูลได้ เป็นต้น

ในการวิจัยนี้เลือกใช้การเก็บข้อมูลความกว้างช่องสัญญาณในหน่วยความจำของเว็บเซิร์ฟเวอร์ เนื่องจากข้อมูลที่ต้องการจัดเก็บมีขนาดเล็ก ไม่จำเป็นต้องเก็บเป็นระยะเวลานาน และต้องการให้ระบบทำงานในลักษณะโปร่งใส และอัตโนมัติ โดยใช้ IPC::Shareable Perl Module [13] ในการทำงานเพื่อให้เว็บเซิร์ฟเวอร์โปรเซสต่าง ๆ สามารถใช้งานข้อมูลร่วมกันได้ การเก็บข้อมูลความกว้างช่องสัญญาณในหน่วยความจำเป็นการช่วยลดการทำงานที่ซ้ำซ้อนในการตรวจสอบความกว้างช่องสัญญาณระหว่างเว็บเซิร์ฟเวอร์ และไคลเอนต์ แต่เนื่องจากความกว้างช่องสัญญาณมีการเปลี่ยนแปลงตามสภาพแวดล้อมในช่วงเวลาต่าง ๆ ทำให้จำเป็นต้องมีวิธีการแก้ไขข้อมูลความกว้างช่องสัญญาณที่ถูกจัดเก็บไว้ในหน่วยความจำให้ถูกต้อง นอกจากนี้หน่วยความจำยังมีขนาดที่จำกัดอีกด้วย ในการวิจัยนี้จึงเลือกใช้วิธีการ Round – trip ข้อมูลความกว้างช่องสัญญาณในหน่วยความจำ โดยกำหนดจำนวนข้อมูลสูงสุดที่สามารถจัดเก็บในหน่วยความจำได้ เมื่อมีข้อมูลเกินกว่าค่าที่กำหนดไว้ระบบจะทำการลบข้อมูลที่มีอายุมากที่สุดออกโดยอัตโนมัติ ด้วยวิธีการนี้ทำให้ค่าความกว้างช่องสัญญาณที่อยู่ในหน่วยความจำสามารถเปลี่ยนแปลงได้ การกำหนดจำนวนข้อมูลสูงสุดที่สามารถจัดเก็บในหน่วยความจำนี้จะขึ้นกับสภาพแวดล้อมในการทำงานของเว็บเซิร์ฟเวอร์ โดยถ้ามีไคลเอนต์ใช้งานเป็นจำนวนมากควรกำหนดให้มีค่าสูงเนื่องจากจะช่วยลดความถี่ในการตรวจสอบความกว้างช่องสัญญาณได้ แต่ในกรณีที่มีจำนวนไคลเอนต์

น้อยควรกำหนดให้มีค่าต่ำ เนื่องจากค่าที่สูงเกินไปจะทำให้ข้อมูลความกว้างช่องสัญญาณในหน่วยความจำไม่มีการเปลี่ยนแปลง หรือมีการเปลี่ยนแปลงช้า ทำให้ข้อมูลที่เก็บไว้ไม่ตรงกับความเป็นจริง ในการทดสอบนี้ได้กำหนดจำนวนข้อมูลสูงสุดที่สามารถจัดเก็บได้เป็น 10 เนื่องจากไคลเอนต์ที่ใช้ทดสอบมีจำนวนจำกัด

3.3. **ค่าความล่าช้า** (Latency)

เป็นเกณฑ์ที่ใช้ตัดสินใจในการแสดงตัวอย่างข้อมูล โดยถ้าค่าความล่าช้าที่คำนวณได้มีค่าน้อยกว่าค่าที่กำหนดไว้จะไม่ทำการแสดงตัวอย่าง แต่ถ้าค่าที่คำนวณได้มีค่ามากกว่าค่าที่กำหนดไว้ระบบจะทำการแสดงตัวอย่างโดยอัตโนมัติ ค่าความล่าช้าสามารถคำนวณได้จากสูตร

$$\text{ค่าความล่าช้า} = \frac{\text{ขนาดของข้อมูล}}{\text{ความกว้างช่องสัญญาณ}}$$

ในการวิจัยนี้ได้ทดสอบโดยกำหนดค่าความล่าช้าเท่ากับ 5

3.4. **การแสดงตัวอย่างของข้อมูล**

จากทฤษฎีที่เกี่ยวข้องในบทที่ 2 แสดงถึงไฟล์ข้อมูลภาพ และภาพเคลื่อนไหวรูปแบบต่าง ๆ ที่นิยมใช้งานในอินเทอร์เน็ต ในการวิจัยนี้เลือกใช้ไฟล์เจเพ็กสำหรับข้อมูลภาพ และเอ็มเพ็กสำหรับภาพเคลื่อนไหว เนื่องจากสนับสนุนจำนวนสีมาก และมีการใช้งานกันอย่างแพร่หลาย [16] [17]

3.4.1. ข้อมูลภาพ

พิจารณาวิธีแสดงตัวอย่างข้อมูลภาพ 3 วิธี ได้แก่ การตัดข้อมูลภาพให้เหลือพื้นที่ 25% (Crop 50% all sides) การย่อความกว้างยาวของข้อมูลภาพให้เหลือ 25% (Geometry 50% all sides) หรือการทำ Thumbnail และการแปลงไฟล์ข้อมูลภาพให้อยู่ในรูปแบบไฟล์เจเพ็กที่มีคุณภาพระดับ 5 (Convert to JPG quality level 5) โดยวิธีการตัดข้อมูลภาพให้เหลือพื้นที่ 25% จะทำการตัดข้อมูลภาพออกด้านละ 50% ซึ่งจะทำให้องค์ประกอบของข้อมูลภาพหายไป แต่โดยทั่วไปแล้วส่วนประกอบที่สำคัญของข้อมูลภาพจะอยู่ตรงส่วนกลาง ดังนั้นตัวอย่างข้อมูลภาพที่ได้รับยังคงสามารถนำไปใช้งานได้ สำหรับวิธีการแปลงไฟล์ข้อมูลภาพให้อยู่ในรูปแบบไฟล์เจเพ็กที่มีคุณภาพระดับ 5 เป็นระดับคุณภาพที่ JPEG image compression FAQ แนะนำให้ใช้สำหรับการแสดงตัวอย่างข้อมูลภาพ [7] โดยระดับคุณภาพ หมายถึง ระดับคุณภาพของไฟล์เจเพ็กที่ได้รับ โดยจะมีค่าอยู่ระหว่าง 0 – 100 ซึ่ง 0 หมายถึง ไฟล์เจเพ็กที่มีคุณภาพแย่ที่สุด และ 100 หมายถึง ไฟล์เจเพ็กที่คุณภาพดีที่สุด

ในการวิจัยนี้จะทำการทดสอบเพื่อเลือกวิธีการแสดงตัวอย่างข้อมูลภาพโดยจะใช้วิธีต่าง ๆ ดังกล่าวข้างต้นกับไฟล์ข้อมูลภาพขนาดใหญ่ และขนาดใหญ่มาก โดยข้อมูลภาพขนาดใหญ่ หมายถึง ไฟล์ข้อมูลภาพที่มีขนาดมากกว่า 70 กิโลไบต์ แต่ไม่เกิน 100 กิโลไบต์ และข้อมูลภาพขนาดใหญ่มาก หมายถึง ไฟล์ข้อมูลภาพที่มีขนาดมากกว่า 100 กิโลไบต์ ด้านล่างแสดงวิธีต่าง ๆ ที่เป็นไปได้ทั้งหมดที่ใช้ในการทดสอบ

1. Crop 50% all sides
2. Geometry 50% all sides
3. Convert to JPG quality level 5
4. Crop 50% all sides และ Geometry 50% all sides
5. Crop 50% all sides และ Convert to JPG quality level 5
6. Geometry 50% all sides และ Crop 50% all sides
7. Geometry 50% all sides และ Convert to JPG quality level 5
8. Convert to JPG quality level 5และ Crop 50% all sides
9. Convert to JPG quality level 5และ Geometry 50% all sides
10. Crop 50% all sides, Geometry 50% all sides และ Convert to JPG quality level 5
11. Crop 50% all sides, Convert to JPG quality level 5 และ Geometry 50% all sides
12. Geometry 50% all sides, Convert to JPG quality level 5 และ Crop 50% all sides
13. Geometry 50% all sides, Crop 50% all sides และ Convert to JPG quality level 5
14. Convert to JPG quality level 5, Crop 50% all sides และ Geometry 50% all sides
15. Convert to JPG quality level 5, Geometry 50% all sides และ Crop 50% all sides

ในการทดสอบใช้โปรแกรม ImageMagick [8] ซึ่งเป็นกลุ่มของเครื่องมือ และ Library ในการจัดการข้อมูลภาพในรูปแบบต่าง ๆ โดยสามารถทำงานผ่าน Command Line หรือผ่านทาง Programming Interface ของภาษาซี ซีพลัส พลัส หรือเพิร์ล

ตารางที่ 3.2 – 3.5 และรูปที่ 3.1 – 3.3 แสดงเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพขนาดใหญ่ และขนาดใหญ่มากด้วยวิธีต่าง ๆ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพขนาดใหญ่ และขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ ตามลำดับ และรูปที่ 3.4 และ 3.5 แสดงเวลาทั้งหมดที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพขนาดใหญ่ และขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ ตามลำดับ โดยค่า Processing time ในตารางที่ 3.2 และ 3.3 ได้จากการหาค่าเฉลี่ยของผลการทดสอบแสดงตัวอย่างข้อมูล ค่า Transmission time ในตารางที่ 3.4 และ 3.5 ได้จากการคำนวณเวลาที่ใช้ในการส่งไฟล์ข้อมูลบนความกว้างช่องสัญญาณขนาดต่าง ๆ และเครื่องหมายบวกในตารางที่ 3.2 – 3.5 และรูปที่ 3.1 – 3.5 หมายถึง ลำดับในการแสดงตัวอย่างข้อมูล

ตารางที่ 3.2 แสดงขนาดไฟล์ และเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพขนาดใหญ่ด้วยวิธีต่าง ๆ

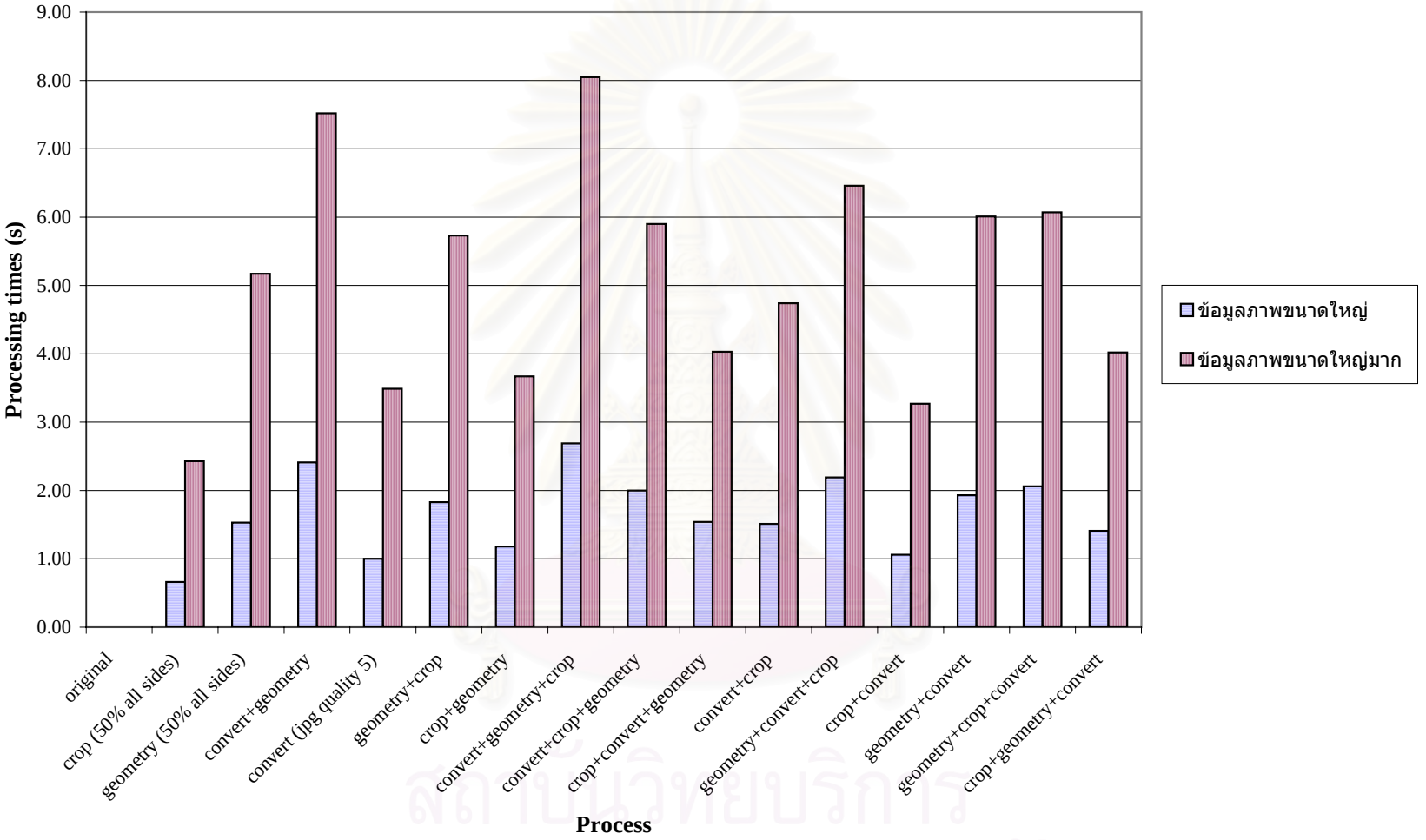
Process	Size			Processing times (s)
	Bytes	Kbytes	%	
Original	70,048	68.41	100.00	0.00
Crop 50% all sides	17,673	17.26	25.23	0.66
Geometry 50% all sides	15,507	15.14	22.14	1.53
Convert + Geometry	13,790	13.47	19.69	2.41
Convert to JPG quality level 5	6,012	5.87	8.58	1.00
Geometry + Crop	4,624	4.52	6.60	1.83
Crop + Geometry	4,550	4.44	6.50	1.18
Convert + Geometry + Crop	4,314	4.21	6.16	2.69
Convert + Crop + Geometry	4,215	4.12	6.02	2.00
Crop + Convert + Geometry	4,194	4.10	5.99	1.54
Convert + Crop	4,016	3.92	5.73	1.51
Geometry + Convert + Crop	2,800	2.73	4.00	2.19
Crop + Convert	1,996	1.95	2.85	1.06
Geometry + Convert	1,875	1.83	2.68	1.93
Geometry + Crop + Convert	762	0.74	1.09	2.06
Crop + Geometry + Convert	757	0.74	1.08	1.41

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 3.3 แสดงขนาดไฟล์ และเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพขนาดใหญ่มากด้วยวิธีต่าง ๆ

Process	Size			Processing times (s)
	Bytes	Kbytes	%	
Original	731,533	714.39	100.00	0.00
Geometry 50% all sides	43,630	42.61	5.96	5.17
Crop 50% all sides	39,168	38.25	5.35	2.43
Convert + Geometry	36,935	36.07	5.05	7.52
Convert + Crop	20,665	20.18	2.82	4.74
Convert to JPG quality level 5	17,302	16.90	2.37	3.49
Crop + Geometry	14,744	14.40	2.02	3.67
Geometry + Crop	14,616	14.27	2.00	5.73
Convert + Crop + Geometry	13,612	13.29	1.86	5.90
Convert + Geometry + Crop	13,590	13.27	1.86	8.05
Geometry + Convert + Crop	12,520	12.23	1.71	6.46
Crop + Convert + Geometry	10,629	10.38	1.45	4.03
Geometry + Convert	8,678	8.47	1.19	6.01
Crop + Convert	8,432	8.23	1.15	3.27
Crop + Geometry + Convert	6,399	6.25	0.87	4.02
Geometry + Crop + Convert	6,393	6.24	0.87	6.07

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



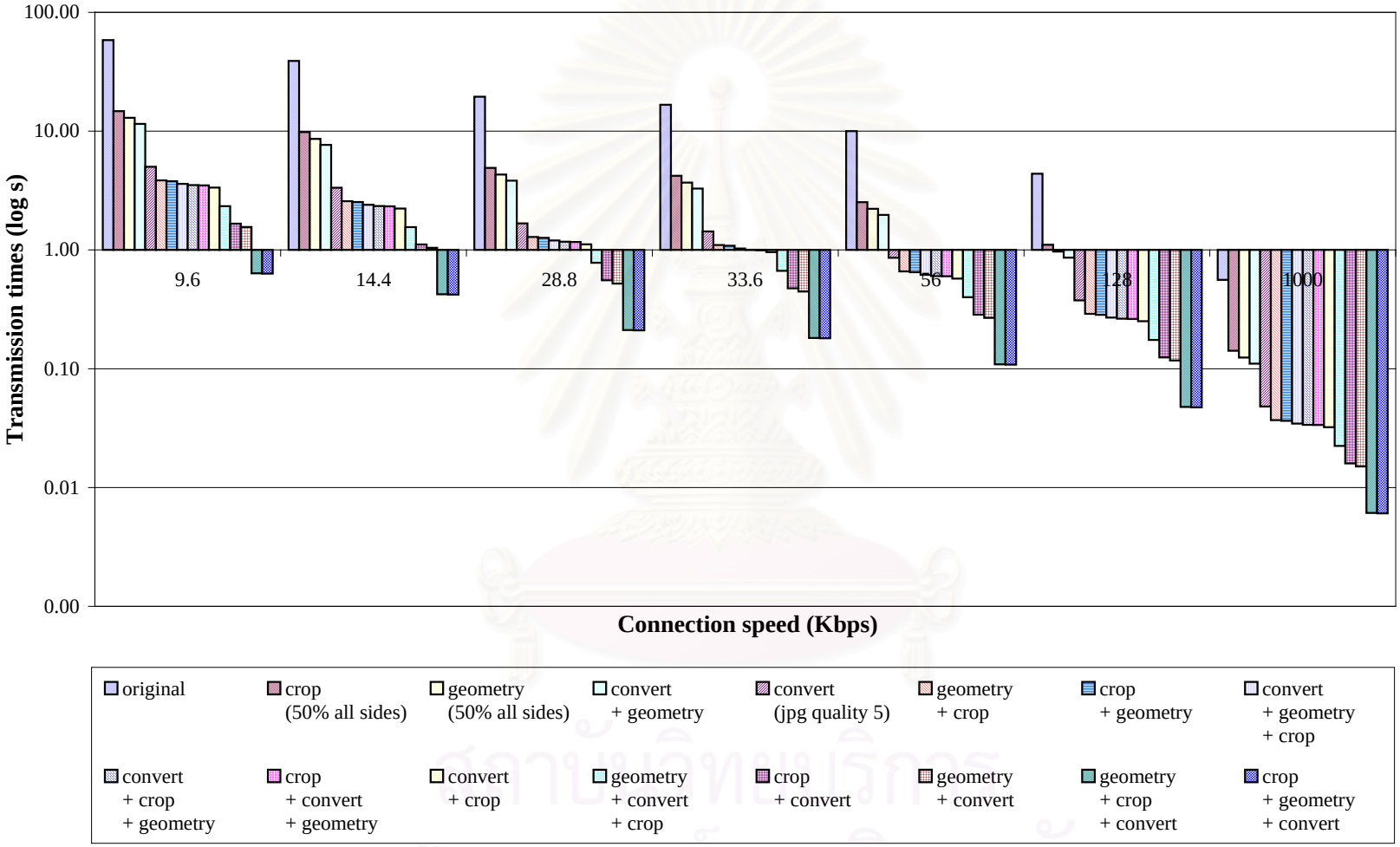
รูปที่ 3.1 แสดงเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพด้วยวิธีต่าง ๆ

ตารางที่ 3.4 แสดงขนาดไฟล์ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ

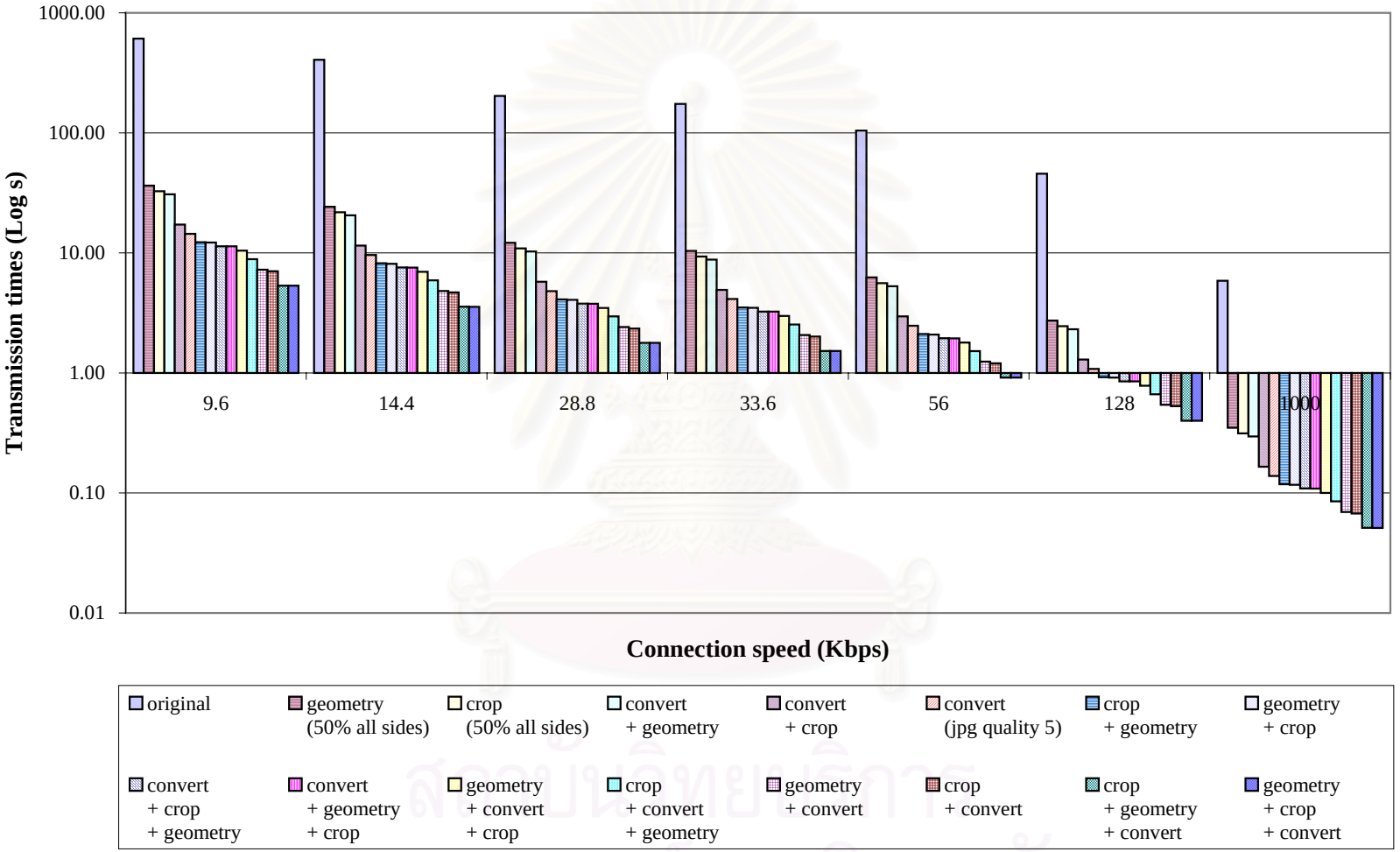
Process	Size			Transmission times (s)						
	bytes	Kbytes	%	9.6	14.4	28.8	33.6	56	128	1000
Original	70,048	68.41	100.00	58.37	38.92	19.46	16.68	10.01	4.38	0.56
Crop 50% all sides	17,673	17.26	25.23	14.73	9.82	4.91	4.21	2.52	1.10	0.14
Geometry 50% all sides	15,507	15.14	22.14	12.92	8.62	4.31	3.69	2.22	0.97	0.12
Convert + Geometry	13,790	13.47	19.69	11.49	7.66	3.83	3.28	1.97	0.86	0.11
Convert to JPG quality level 5	6,012	5.87	8.58	5.01	3.34	1.67	1.43	0.86	0.38	0.05
Geometry + Crop	4,624	4.52	6.60	3.85	2.57	1.28	1.10	0.66	0.29	0.04
Crop + Geometry	4,550	4.44	6.50	3.79	2.53	1.26	1.08	0.65	0.28	0.04
Convert + Geometry + Crop	4,314	4.21	6.16	3.60	2.40	1.20	1.03	0.62	0.27	0.03
Convert + Crop + Geometry	4,215	4.12	6.02	3.51	2.34	1.17	1.00	0.60	0.26	0.03
Crop + Convert + Geometry	4,194	4.10	5.99	3.50	2.33	1.17	1.00	0.60	0.26	0.03
Convert + Crop	4,016	3.92	5.73	3.35	2.23	1.12	0.96	0.57	0.25	0.03
Geometry + Convert + Crop	2,800	2.73	4.00	2.33	1.56	0.78	0.67	0.40	0.18	0.02
Crop + Convert	1,996	1.95	2.85	1.66	1.11	0.55	0.48	0.29	0.12	0.02
Geometry + Convert	1,875	1.83	2.68	1.56	1.04	0.52	0.45	0.27	0.12	0.02
Geometry + Crop + Convert	762	0.74	1.09	0.64	0.42	0.21	0.18	0.11	0.05	0.01
Crop + Geometry + Convert	757	0.74	1.08	0.63	0.42	0.21	0.18	0.11	0.05	0.01

ตารางที่ 3.5 แสดงขนาดไฟล์ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ

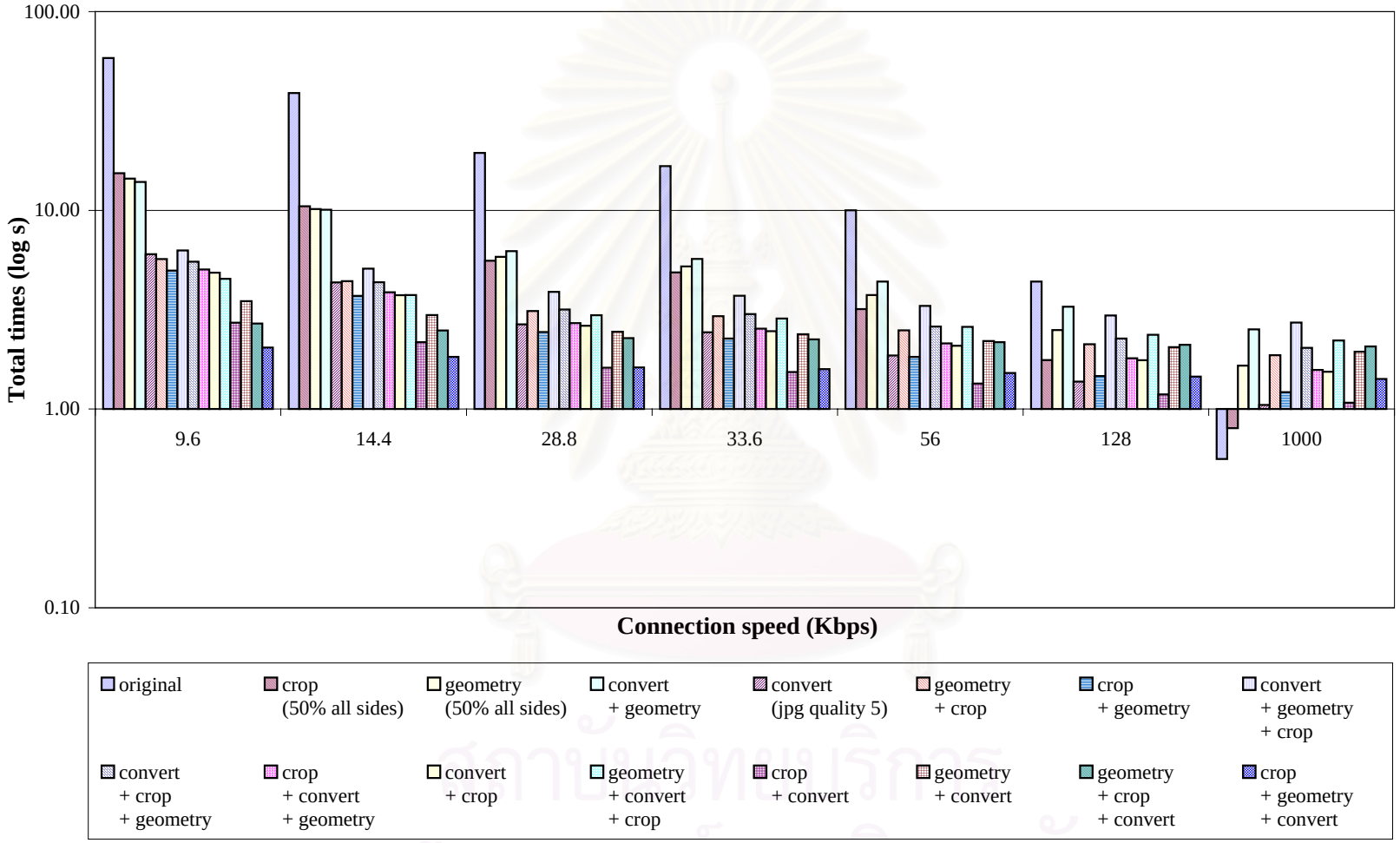
Process	Size			Ttransmission times (s)						
	bytes	Kbytes	%	9.6	14.4	28.8	33.6	56	128	1000
Original	731,533	714.39	100.00	609.61	406.41	203.20	174.17	104.50	45.72	5.85
Geometry 50% all sides	43,630	42.61	5.96	36.36	24.24	12.12	10.39	6.23	2.73	0.35
Crop 50% all sides	39,168	38.25	5.35	32.64	21.76	10.88	9.33	5.60	2.45	0.31
Convert + Geometry	36,935	36.07	5.05	30.78	20.52	10.26	8.79	5.28	2.31	0.30
Convert + Crop	20,665	20.18	2.82	17.22	11.48	5.74	4.92	2.95	1.29	0.17
Convert to JPG quality level 5	17,302	16.90	2.37	14.42	9.61	4.81	4.12	2.47	1.08	0.14
Crop + Geometry	14,744	14.40	2.02	12.29	8.19	4.10	3.51	2.11	0.92	0.12
Geometry + Crop	14,616	14.27	2.00	12.18	8.12	4.06	3.48	2.09	0.91	0.12
Convert + Crop + Geometry	13,612	13.29	1.86	11.34	7.56	3.78	3.24	1.94	0.85	0.11
Convert + Geometry + Crop	13,590	13.27	1.86	11.33	7.55	3.78	3.24	1.94	0.85	0.11
Geometry + Convert + Crop	12,520	12.23	1.71	10.43	6.96	3.48	2.98	1.79	0.78	0.10
Crop + Convert + Geometry	10,629	10.38	1.45	8.86	5.91	2.95	2.53	1.52	0.66	0.09
Geometry + Convert	8,678	8.47	1.19	7.23	4.82	2.41	2.07	1.24	0.54	0.07
Crop + Convert	8,432	8.23	1.15	7.03	4.68	2.34	2.01	1.20	0.53	0.07
Crop + Geometry + Convert	6,399	6.25	0.87	5.33	3.56	1.78	1.52	0.91	0.40	0.05
Geometry + Crop + Convert	6,393	6.24	0.87	5.33	3.55	1.78	1.52	0.91	0.40	0.05



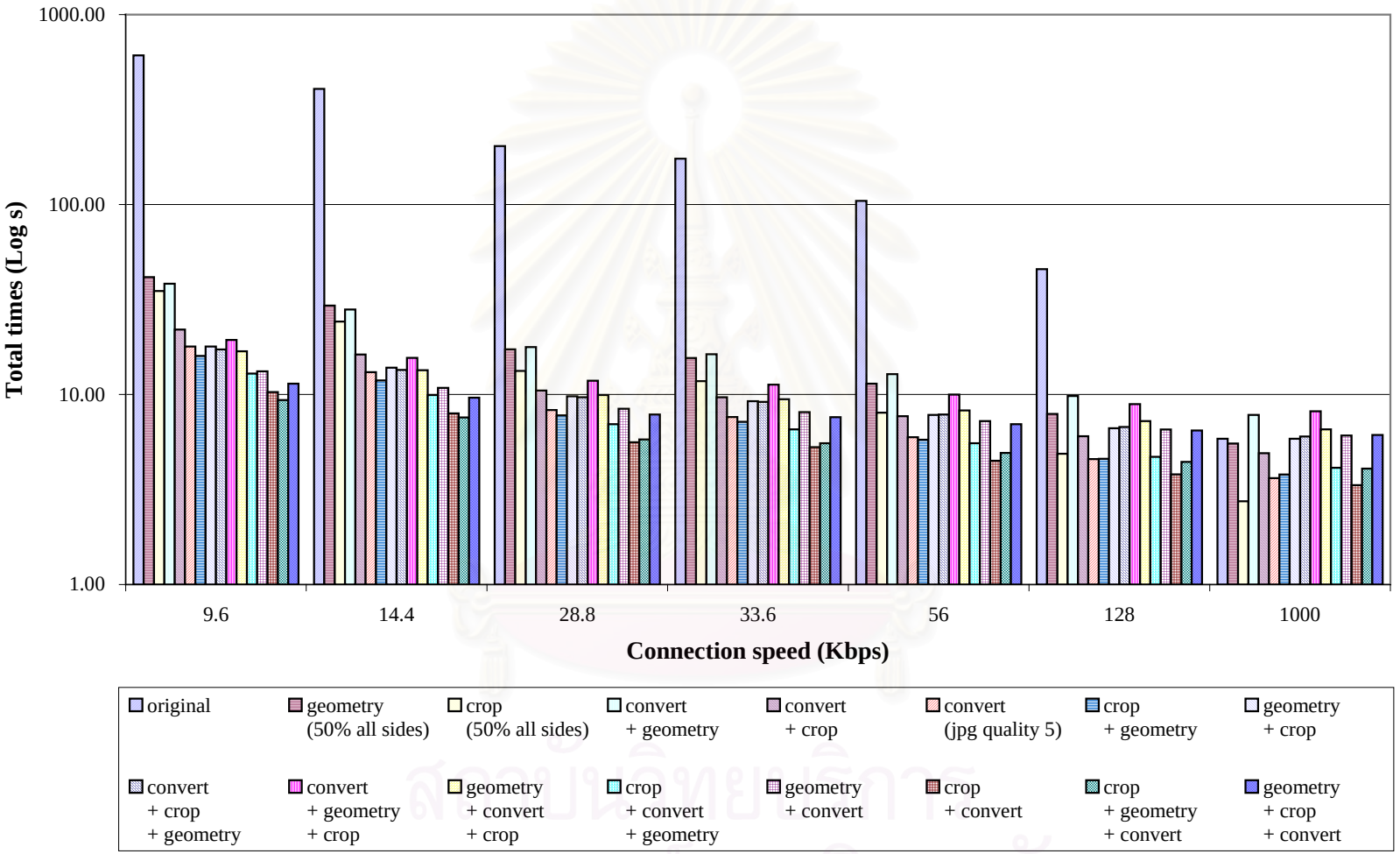
รูปที่ 3.2 แสดงเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ



รูปที่ 3.3 แสดงเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ



รูปที่ 3.4 แสดงเวลาที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ



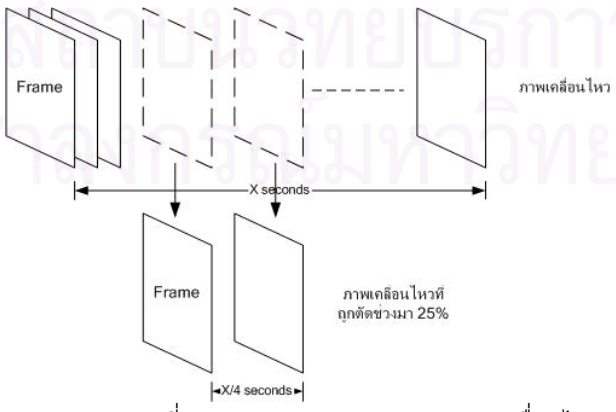
รูปที่ 3.5 แสดงเวลาที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ

จากรูปที่ 3.4 และ 3.5 สรุปได้ว่าวิธีการตัดข้อมูลภาพให้เหลือพื้นที่ 25% และแปลงไฟล์ข้อมูลภาพให้อยู่ในรูปแบบไฟล์เจเพ็กที่มีคุณภาพระดับ 5 (Crop + Convert) มีประสิทธิภาพดีที่สุด เนื่องจากใช้เวลาทั้งหมดน้อยที่สุด และมีขั้นตอนการทำงานน้อยกว่าวิธีการตัดข้อมูลภาพให้เหลือพื้นที่ 25% การย่อความกว้างยาวของข้อมูลภาพให้เหลือ 25% และแปลงไฟล์ข้อมูลภาพให้อยู่ในรูปแบบไฟล์เจเพ็กที่มีคุณภาพระดับ 5 (Crop + Geometry + Convert) ซึ่งทำให้สามารถ Scalable ได้เมื่อระบบมี Load เพิ่มมากขึ้น นอกจากนี้จากรูปยังแสดงให้เห็นด้วยว่าการวิจัยนี้ไม่สามารถใช้งานได้กับการเชื่อมต่อที่มีความกว้างช่องสัญญาณมากกว่า 128 Kbps ได้เนื่องจากเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพจะมากกว่าเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพให้แก่ไคลเอนต์

3.4.2. ข้อมูลภาพเคลื่อนไหว

พิจารณาวิธีแสดงตัวอย่างข้อมูลภาพเคลื่อนไหว 2 วิธี ได้แก่ การตัดช่วงของภาพเคลื่อนไหวมา 25% (Split 25%) แสดงดังรูปที่ 3.6 และการลดขนาดของกรอบภาพเคลื่อนไหวให้เหลือ 160 x 128 จุด (Geometry) โดยใช้โปรแกรม mpgtx [9] ในการตัดช่วงของภาพเคลื่อนไหว และ FFmpeg [10] ในการลดขนาดของกรอบภาพเคลื่อนไหว เนื่องจาก mpgtx สนับสนุนการตัด และรวมช่วงของภาพเคลื่อนไหวตามมาตรฐานเอ็มเพ็ก-1 เท่านั้น ดังนั้นในการวิจัยนี้จะใช้ข้อมูลภาพเคลื่อนไหวตามมาตรฐานเอ็มเพ็ก-1 ขนาดใหญ่ และขนาดใหญ่มาก ในการทดสอบ โดยข้อมูลภาพเคลื่อนไหวขนาดใหญ่ หมายถึง ไฟล์ข้อมูลภาพเคลื่อนไหวที่มีขนาดมากกว่า 700 กิโลไบต์ แต่ไม่เกิน 1 เมกะไบต์ และข้อมูลภาพเคลื่อนไหวขนาดใหญ่มาก หมายถึง ไฟล์ข้อมูลภาพเคลื่อนไหวที่มีขนาดมากกว่า 1 เมกะไบต์ ด้านล่างแสดงวิธีต่าง ๆ ทั้งหมดที่ใช้ในการทดสอบ

1. Split 25%
2. Geometry
3. Split 25% และ Geometry
4. Geometry และ Spilt 25%



รูปที่ 3.6 แสดงการตัดช่วงของภาพเคลื่อนไหวมา 25%

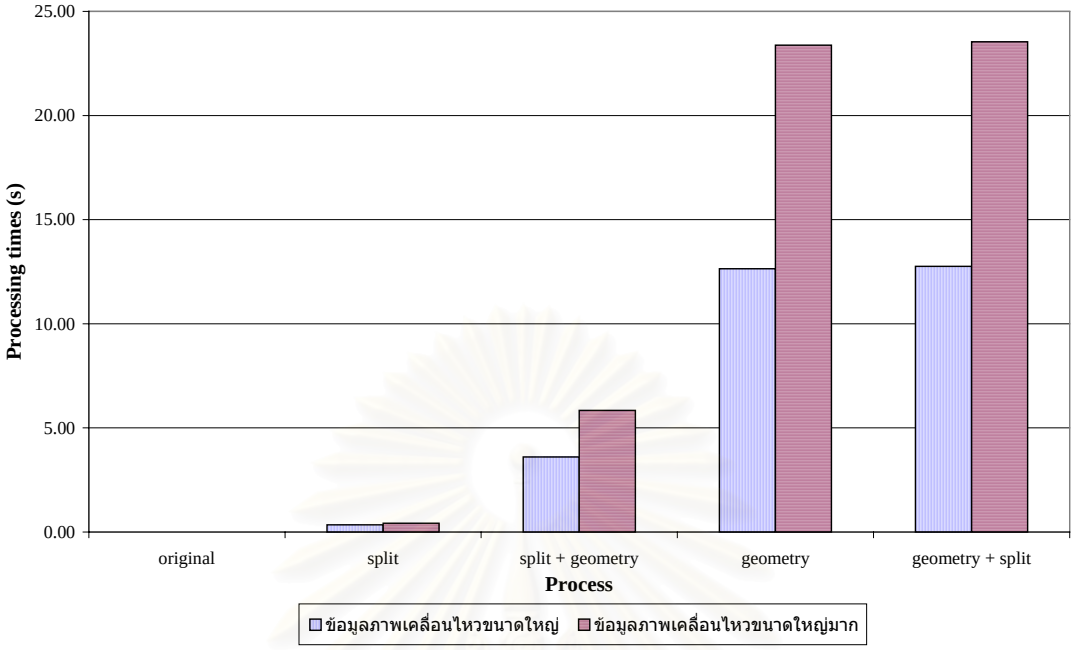
ตารางที่ 3.6 – 3.9 และรูปที่ 3.7 – 3.9 แสดงเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่ และขนาดใหญ่มากด้วยวิธีต่าง ๆ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่ และขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ ตามลำดับ และรูปที่ 3.10 และ 3.11 แสดงเวลาทั้งหมดที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่ และขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ ตามลำดับ โดยค่า Processing time ในตารางที่ 3.6 และ 3.7 ได้จากการหาค่าเฉลี่ยของผลการทดสอบแสดงตัวอย่างข้อมูล ค่า Transmission time ในตารางที่ 3.8 และ 3.9 ได้จากการคำนวณเวลาที่ใช้ในการส่งไฟล์ข้อมูลบนความกว้างช่องสัญญาณขนาดต่าง ๆ และเครื่องหมายบวกในตารางที่ 3.6 – 3.9 และรูปที่ 3.7 – 3.11 หมายถึงลำดับในการแสดงตัวอย่างข้อมูล

ตารางที่ 3.6 แสดงขนาดไฟล์ และเวลาที่ใช้ในการแสดง
ตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่ด้วยวิธีต่าง ๆ

Process	Size			Processing times (s)
	Bytes	Kbytes	%	
Original	738,502	721.19	100.00%	0.00
Geometry	299,008	292.00	40.49%	12.65
Split	220,813	215.64	29.90%	0.35
Geometry + Split	93,645	91.45	12.68%	12.76
Split + Geometry	88,064	86.00	11.92%	3.60

ตารางที่ 3.7 แสดงขนาดไฟล์ และเวลาที่ใช้ในการแสดง
ตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่มากด้วยวิธีต่าง ๆ

Process	Size			Processing times (s)
	Bytes	Kbytes	%	
Original	3,231,625	3,155.88	100.00%	0.00
Split	828,632	809.21	25.64%	0.43
Geometry	526,336	514.00	16.29%	23.38
Split + Geometry	151,552	148.00	4.69%	5.84
Geometry + Split	145,599	142.19	4.51%	23.54



รูปที่ 3.7 แสดงเวลาที่ใช้ในการแสดงตัวอย่างข้อมูลภาพเคลื่อนไหวด้วยวิธีต่าง ๆ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

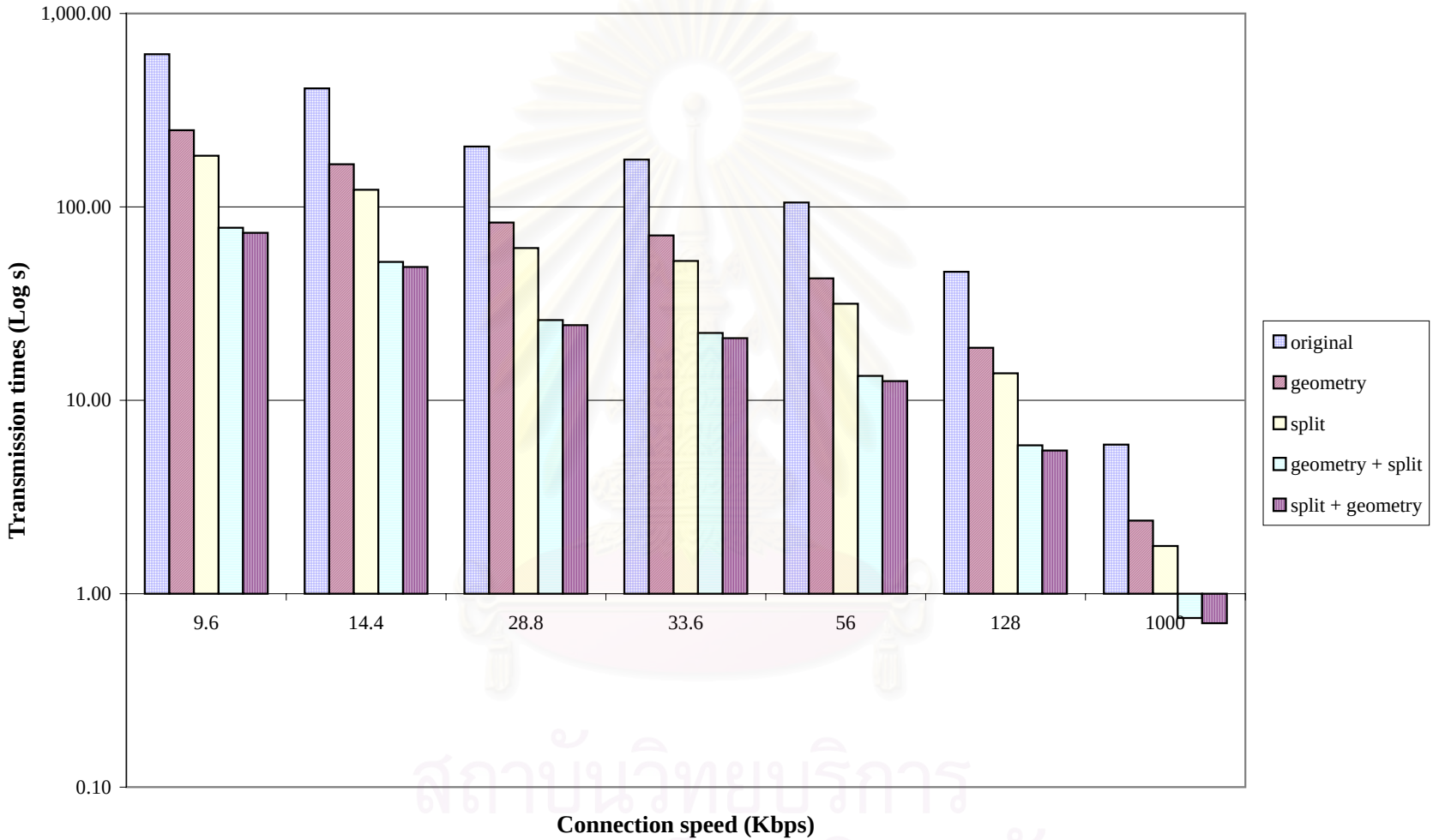
ตารางที่ 3.8 แสดงขนาดไฟล์ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ

Process	Size			Transmission times (s)						
	bytes	Kbytes	%	9.6	14.4	28.8	33.6	56	128	1000
Original	738,502	721.19	100.00%	615.42	410.28	205.14	175.83	105.50	46.16	5.91
Geometry	299,008	292.00	40.49%	249.17	166.12	83.06	71.19	42.72	18.69	2.39
Split	220,813	215.64	29.90%	184.01	122.67	61.34	52.57	31.54	13.80	1.77
Geometry + Split	93,645	91.45	12.68%	78.04	52.03	26.01	22.30	13.38	5.85	0.75
Split + Geometry	88,064	86.00	11.92%	73.39	48.92	24.46	20.97	12.58	5.50	0.70

ตารางที่ 3.9 แสดงขนาดไฟล์ และเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ

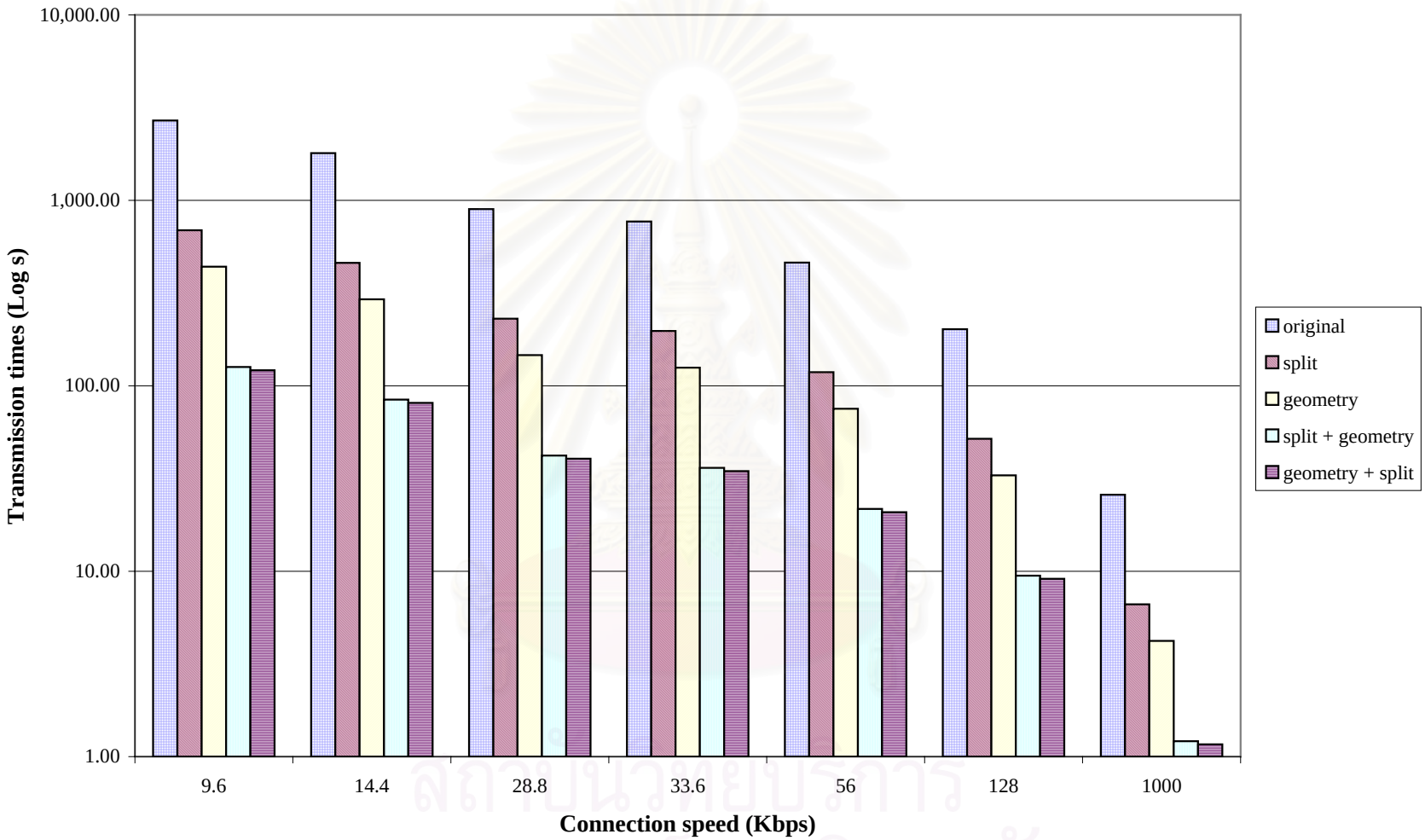
Process	Size			Transmission times (s)						
	bytes	Kbytes	%	9.6	14.4	28.8	33.6	56	128	1000
Original	3,231,625	3,155.88	100.00%	2,693.02	1,795.35	897.67	769.43	461.66	201.98	25.85
Split	828,632	809.21	25.64%	690.53	460.35	230.18	197.29	118.38	51.79	6.63
Geometry	526,336	514.00	16.29%	438.61	292.41	146.20	125.32	75.19	32.90	4.21
Split + Geometry	151,552	148.00	4.69%	126.29	84.20	42.10	36.08	21.65	9.47	1.21
Geometry + Split	145,599	142.19	4.51%	121.33	80.89	40.44	34.67	20.80	9.10	1.16

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

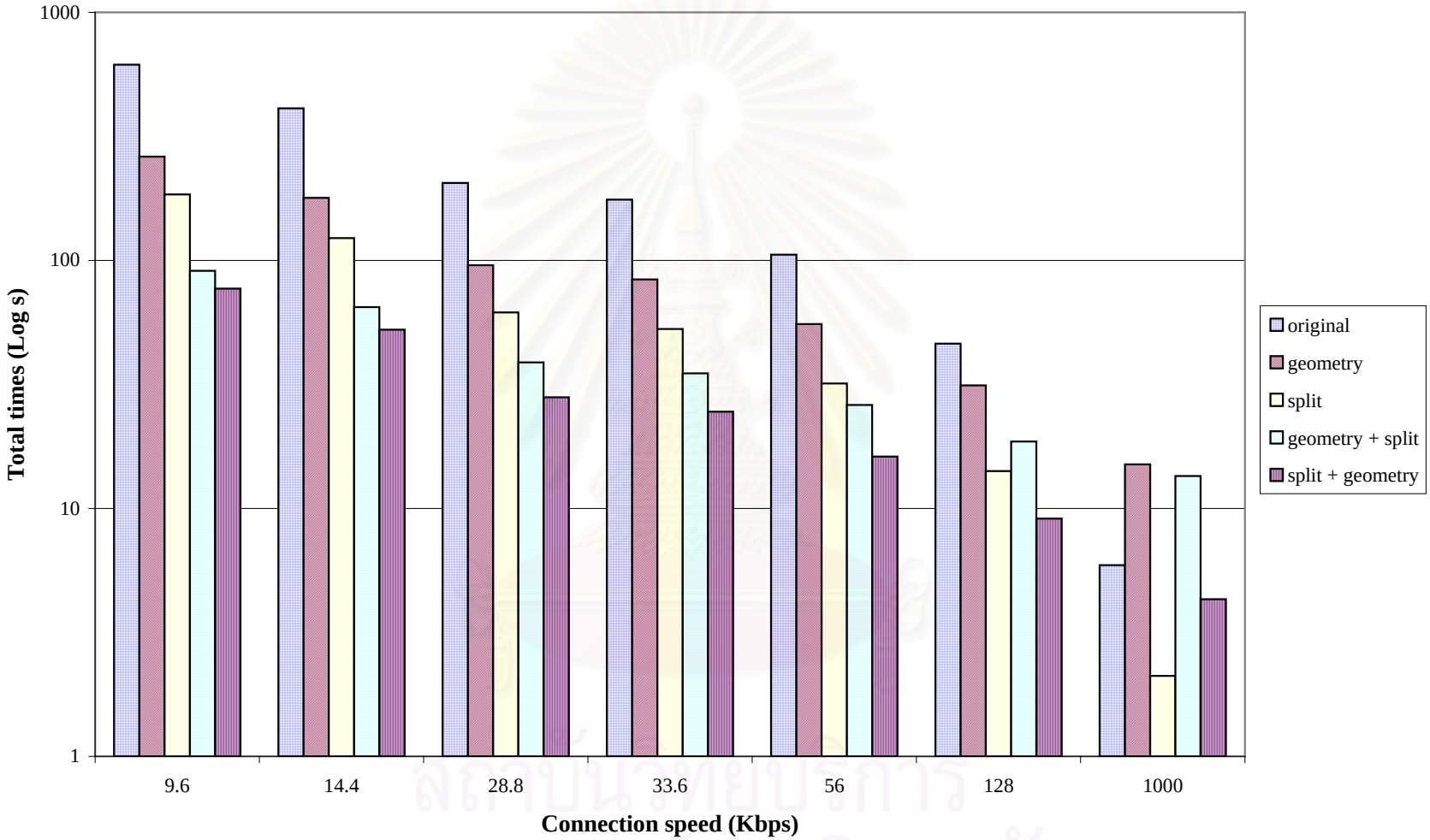


รูปที่ 3.8 แสดงเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ

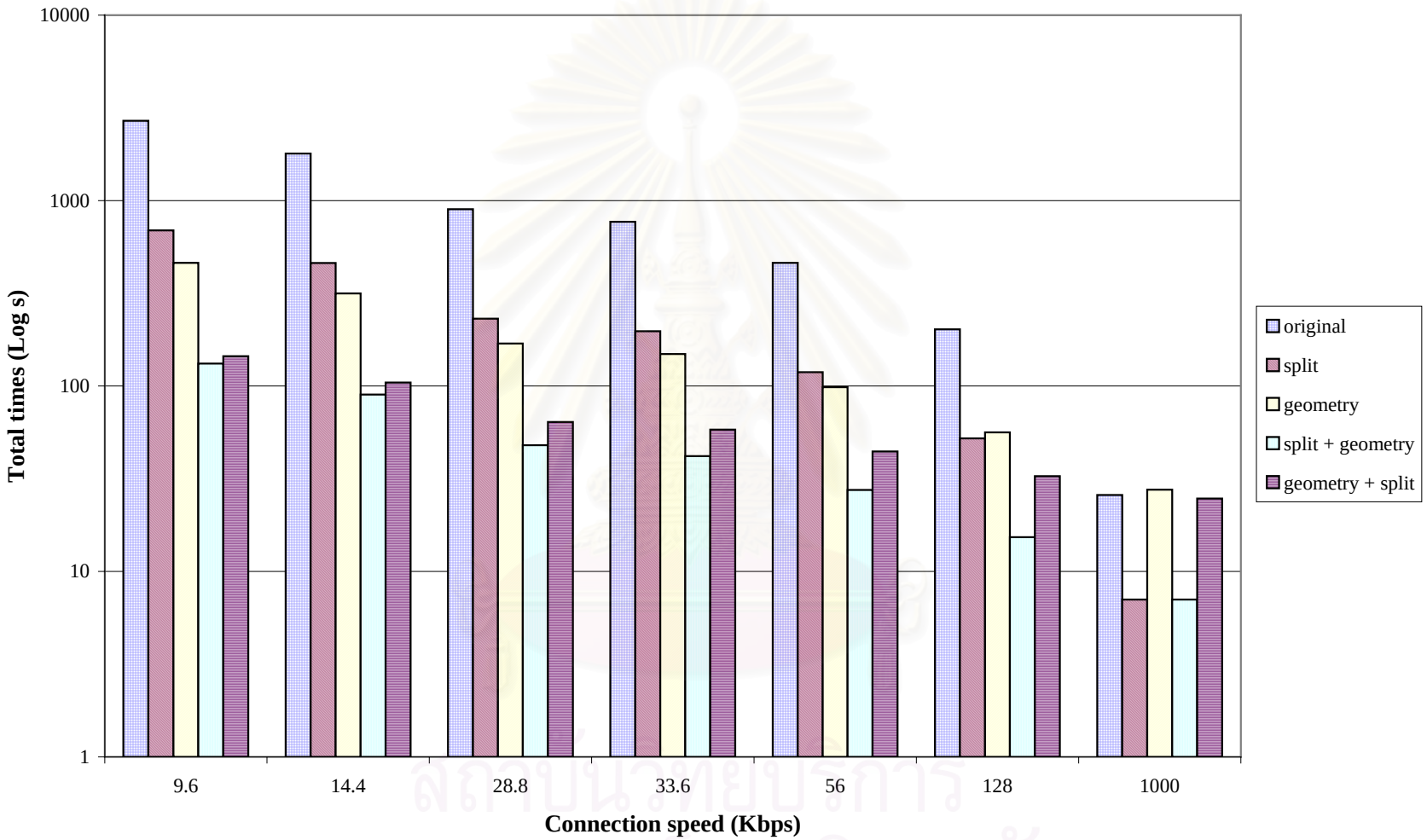
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.9 แสดงเวลาที่ใช้ในการส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ



รูปที่ 3.10 แสดงเวลาที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ

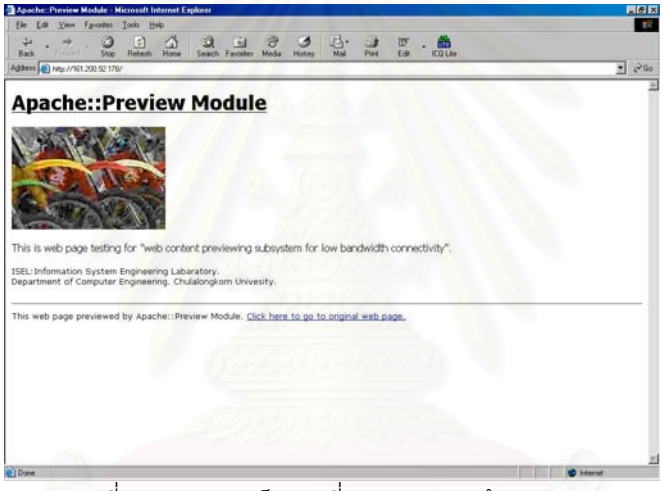


รูปที่ 3.11 แสดงเวลาที่ใช้ในการแสดงตัวอย่าง และส่งตัวอย่างข้อมูลภาพเคลื่อนไหวขนาดใหญ่มากบนการเชื่อมต่อที่มีความกว้างช่องสัญญาณขนาดต่าง ๆ

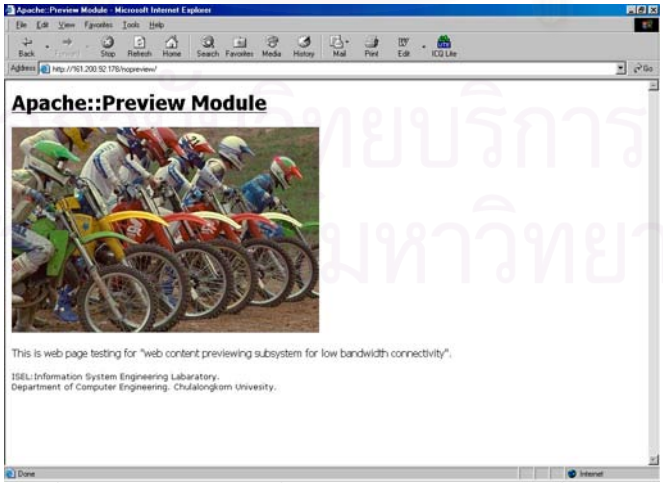
จากรูปที่ 3.10 และ 3.11 สรุปได้ว่าวิธีการตัดช่วงของภาพเคลื่อนไหวมา 25% และลดขนาดของกรอบภาพเคลื่อนไหวให้เหลือ 160 x 128 จุด (Split + Geometry) มีประสิทธิภาพดีที่สุด เนื่องจากใช้เวลารวมทั้งหมดน้อยที่สุด

3.5. **อินเตอร์เฟส**

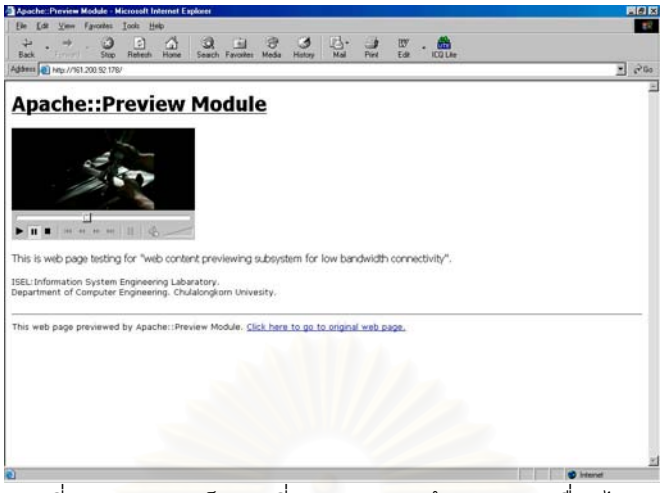
เนื่องจากต้องการให้การแสดงตัวอย่างข้อมูล และการแสดงข้อมูลในรูปแบบที่สมบูรณ์รบกวนการทำงานปกติของผู้ใช้งานน้อยที่สุด ดังนั้นการวิจัยนี้จึงใช้วิธีการแทนที่ข้อมูลภาพ หรือข้อมูลภาพเคลื่อนไหว ด้วยตัวอย่างข้อมูลภาพ หรือภาพเคลื่อนไหวเท่านั้น และทำการแทรก Link เพื่อใช้ในการร้องขอข้อมูลในรูปแบบที่สมบูรณ์ของเว็บเพจหน้านั้น ๆ ไว้ส่วนท้าย (Footer) ของหน้าเว็บ แสดงดังรูปที่ 3.12 – 3.15 สำหรับข้อมูลภาพและภาพเคลื่อนไหวตามลำดับ



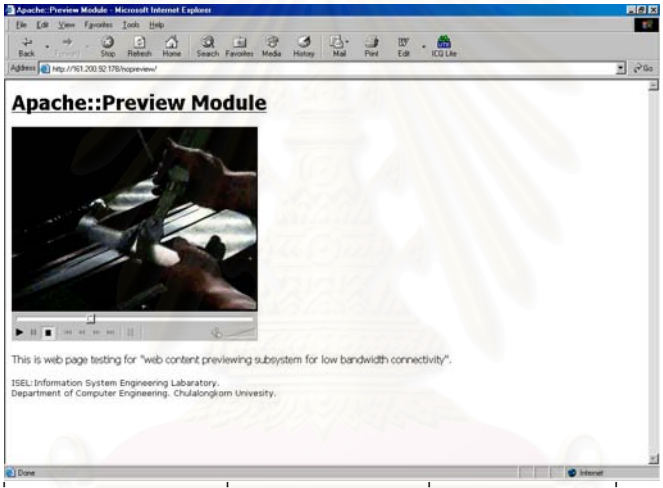
รูปที่ 3.12 แสดงเว็บเพจที่แสดงตัวอย่างข้อมูลภาพ



รูปที่ 3.13 แสดงเว็บเพจที่แสดงข้อมูลภาพในรูปแบบที่สมบูรณ์



รูปที่ 3.14 แสดงเว็บเพจที่แสดงตัวอย่างข้อมูลภาพเคลื่อนไหว

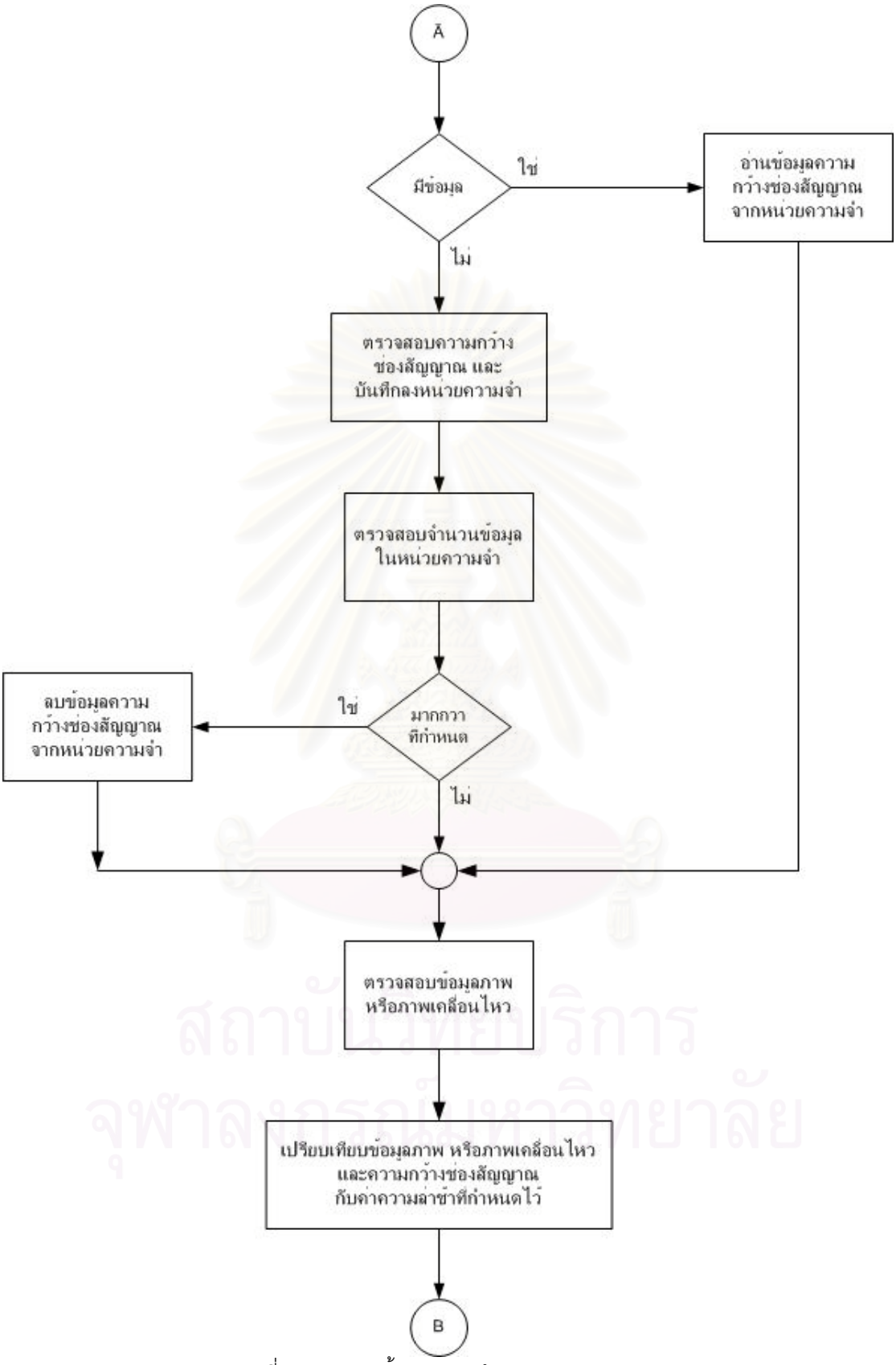


รูปที่ 3.15 แสดงเว็บเพจที่แสดงข้อมูลภาพเคลื่อนไหวในรูปแบบที่สมบูรณ์

3.6. **เครื่องมือที่ใช้ในการวิจัย**

ในการวิจัยนี้ผู้วิจัยได้พัฒนาโมดูลสำหรับอาปาเช่ โดยใช้ภาษาเพิร์ล [3] [4] เนื่องจากภาษาเพิร์ล เป็นภาษาสคริปต์ (Script) ทำให้การพัฒนาสามารถทำได้ง่าย และสะดวกกว่าการใช้ภาษาซีมาก นอกจากนี้ยังมีประสิทธิภาพในการทำงานอีกด้วย โมดูลที่พัฒนาขึ้นนี้จะทำงานในขั้นตอนการสร้างเอกสารเพื่อตอบสนองคำร้องขอจากไคลเอนต์แสดงดังรูปที่ 2.2 และแผนผังแสดงขั้นตอนการทำงานของระบบแสดงดังรูปที่ 3.16

รูปที่ 3.16 แสดงขั้นตอนการทำงานของระบบ



รูปที่ 3.16 แสดงขั้นตอนการทำงานของระบบ (ต่อ)

รูปที่ 3.16 แสดงขั้นตอนการทำงานของระบบ (ต่อ)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4
ผลการวิจัย

จากการศึกษาเพื่อออกแบบ และพัฒนาระบบแสดงตัวอย่างข้อมูลจากเว็บเพจสำหรับการเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำนั้น สามารถสรุปผลการวิจัยได้ดังนี้

4.1. การตรวจสอบความกว้างช่องสัญญาณ

จากการศึกษาพบว่า การตรวจสอบความกว้างช่องสัญญาณที่เหมาะสม คือ การใช้แอปเพล็ต เนื่องจากทำงานโดยใช้โพรโทคอลเอชทีทีพี ทำให้ไม่มีข้อจำกัดในเรื่องของไฟล์วอล์ และโปรแกรมบราวเซอร์ที่นิยมในปัจจุบันมีการติดตั้งจาวาอินเทอร์พรีเตอร์มาด้วยอยู่แล้ว

4.2. การเก็บข้อมูลความกว้างช่องสัญญาณ

จากการศึกษาพบว่า การเก็บข้อมูลความกว้างช่องสัญญาณที่เหมาะสม คือ การเก็บในหน่วยความจำของเว็บเซิร์ฟเวอร์ และใช้วิธีการ Round – trip ข้อมูลโดยกำหนดจำนวนข้อมูลสูงสุดที่สามารถจัดเก็บในหน่วยความจำ เนื่องจากข้อมูลความกว้างช่องสัญญาณมีขนาดเล็ก ไม่จำเป็นต้องเก็บเป็นระยะเวลานาน และมีการเปลี่ยนแปลงตามสภาพแวดล้อมในช่วงเวลาต่าง ๆ

4.3. ค่าความล่าช้า

จากการศึกษาพบว่า ค่าความล่าช้าที่ดี คือ ค่าที่น้อย เนื่องจากแสดงว่ามีความล่าช้าในการส่งข้อมูลระหว่างเว็บเซิร์ฟเวอร์ และไคลเอนต์น้อย ดังนั้นจึงไม่จำเป็นต้องทำการแสดงตัวอย่างข้อมูล ค่าความล่าช้าที่มากแสดงว่ามีความล่าช้าในการส่งข้อมูลระหว่างเว็บเซิร์ฟเวอร์ และไคลเอนต์ จึงจำเป็นต้องมีการแสดงตัวอย่างข้อมูล

4.4. การแสดงตัวอย่างข้อมูล

จากการศึกษาพบว่า การแสดงตัวอย่างข้อมูลภาพที่เหมาะสม คือ การตัดข้อมูลภาพให้เหลือพื้นที่ 25% แล้วแปลงไฟล์ข้อมูลภาพให้อยู่ในรูปแบบไฟล์เจเพ็กที่มีคุณภาพระดับ 5 เนื่องจากใช้เวลาทั้งหมดน้อยที่สุด และมีขั้นตอนการทำงานน้อยอีกด้วย โดยสามารถลดปริมาณข้อมูลได้ 97.15% สำหรับข้อมูลภาพขนาดทั่วไป และ 98.85% สำหรับข้อมูลภาพขนาดใหญ่

สำหรับการแสดงตัวอย่างข้อมูลภาพเคลื่อนไหวที่เหมาะสม คือ การตัดช่วงของภาพเคลื่อนไหวมา 25% แล้วลดขนาดของกรอบภาพเคลื่อนไหวให้เหลือ 160 x 128 จุด เนื่องจากใช้เวลาทั้งหมดน้อยที่สุดเช่นกัน โดยสามารถลดปริมาณข้อมูลได้ 88.08% สำหรับข้อมูลภาพเคลื่อนไหวขนาดทั่วไป และ 95.49% สำหรับข้อมูลภาพเคลื่อนไหวขนาดใหญ่

4.5. ระบบแสดงตัวอย่างข้อมูลจากเว็บเพจ

จากการศึกษาพบว่า การพัฒนาบริการแสดงตัวอย่างข้อมูลสำหรับอาปาเช่บนยูนิกซ์แพลตฟอร์มที่เหมาะสม คือ การใช้ Apache Perl Module เนื่องจากการใช้ Server API หรือ Apache Module จะสร้างบริการที่ทำงานในลักษณะโปร่งใส และอัตโนมัติกับทุกเว็บเพจในเว็บเซิร์ฟเวอร์ได้ และภาษาเพิร์ลจะช่วยทำให้การพัฒนาทำได้สะดวก และรวดเร็วขึ้น เนื่องจากเป็นภาษาที่มีความยืดหยุ่นสูง และเป็นที่นิยมใช้งานกันอย่างแพร่หลาย

4.6. ข้อมูลตัวอย่าง

ผลจากการศึกษานี้ทำให้ได้ข้อมูลตัวอย่างของการแสดงตัวอย่างข้อมูลภาพ และภาพเคลื่อนไหว ดังตัวอย่างต่อไปนี้

ข้อมูลภาพเจเพ็กขนาด 480 x 320 จุด 68.41 กิโลไบต์

ตัวอย่างข้อมูลภาพเจเพ็กคุณภาพระดับ 5 ขนาด 240 x 160 จุด 1.95 กิโลไบต์

ข้อมูลภาพเคลื่อนไหวเอ็มเพ็ก-1 ระยะเวลา 6 วินาที ขนาด 320 x 240 จุด 721.19 กิโลไบต์

ข้อมูลภาพเคลื่อนไหวเอ็มเพ็ก-1 ระยะเวลา 2 วินาที ขนาด 160 x 128 จุด 86.00 กิโลไบต์

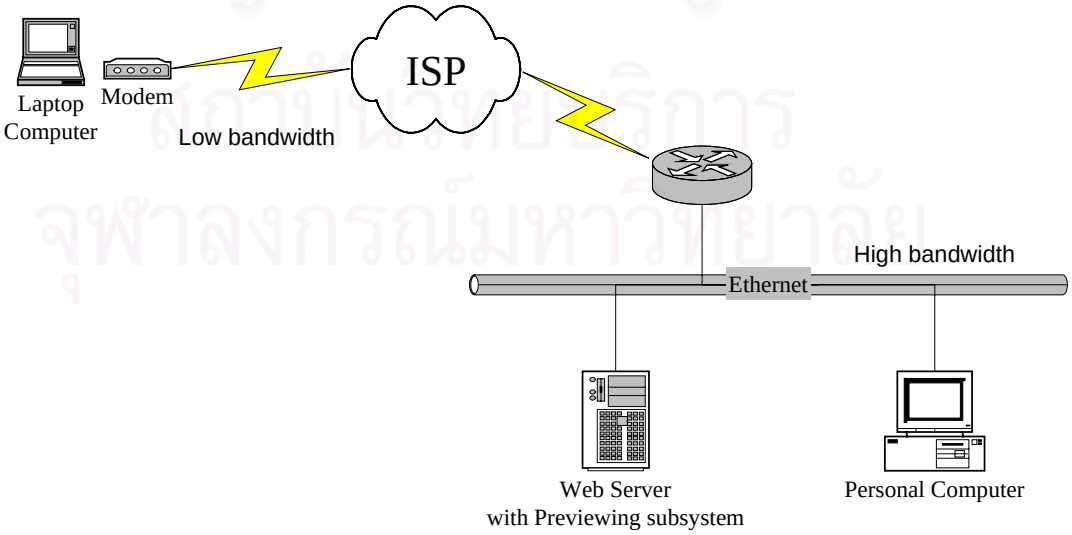
4.7. สภาพแวดล้อมของเครื่องที่ใช้พัฒนา

ในการวิจัยนี้ใช้เครื่องคอมพิวเตอร์ 1 เครื่องในการพัฒนาระบบแสดงตัวอย่างข้อมูล โดยมีรายละเอียดดังนี้

- เครื่องคอมพิวเตอร์แบบพีซี Pentium MMX ความเร็ว 233 เมกะเฮิรตซ์
- หน่วยความจำขนาด 128 เมกะไบต์
- ฮาร์ดดิสก์ขนาด 20 กิกะไบต์
- ใช้ระบบปฏิบัติการลีนุกซ์แรดเฮท 7.3
- ใช้อาปาเช่ 1.3.27 และ mod_perl 1.27

4.8. การทดสอบระบบ

ในการวิจัยนี้ได้ทดสอบระบบโดยให้ไคลเอนต์ทำการร้องขอข้อมูลภาพ และภาพเคลื่อนไหว บนเว็บเซิร์ฟเวอร์ ผ่าน Ethernet และ Dial-up line โดยเปรียบเทียบเวลาที่ใช้ทั้งหมดตั้งแต่คำร้องขอแรกของเว็บเพจถึงเว็บเซิร์ฟเวอร์ จนกระทั่งเว็บเซิร์ฟเวอร์ส่งข้อมูลสุดท้ายของเว็บเพ็จนั้น ๆ ตอบสนองต่อคำร้องขอที่ได้รับ และในระหว่างการทดสอบจะจำกัดให้มีเว็บเซิร์ฟเวอร์เพียงโปรเซสเดียวเท่านั้น รูปที่ 4.1 แสดงการทดสอบระบบแสดงตัวอย่างข้อมูลจากเว็บเพจ และตารางที่ 4.1 และ 4.2 แสดงผลการทดสอบที่ได้รับ



รูปที่ 4.1 แสดงการทดสอบระบบแสดงตัวอย่างข้อมูลจากเว็บเพจ

ตารางที่ 4.1 เปรียบเทียบเวลาที่ใช้ทั้งหมดในการแสดงตัวอย่าง และแสดงข้อมูลแบบสมบูรณ์ของข้อมูลภาพ

ความกว้างช่องสัญญาณ	เวลาที่ใช้แสดงข้อมูลภาพ (s)			
	ขนาดใหญ่		ขนาดใหญ่มาก	
	แสดงตัวอย่าง	แบบสมบูรณ์	แสดงตัวอย่าง	แบบสมบูรณ์
33.6 Kbps	7.95	19.04	10.28	216.52
10 Mbps	6.98	0.07	7.12	0.77

ตารางที่ 4.2 เปรียบเทียบเวลาที่ใช้ทั้งหมดในการแสดงตัวอย่าง
และแสดงข้อมูลแบบสมบูรณ์ของข้อมูลภาพเคลื่อนไหว

ความกว้างช่องสัญญาณ	เวลาที่ใช้แสดงข้อมูลภาพเคลื่อนไหว (s)			
	ขนาดใหญ่		ขนาดใหญ่มาก	
	แสดงตัวอย่าง	แบบสมบูรณ์	แสดงตัวอย่าง	แบบสมบูรณ์
33.6 Kbps	16.92	240.51	42.25	N/A
10 Mbps	9.58	1.13	12.41	4.28

จากตารางที่ 4.1 และ 4.2 สามารถสรุปได้ว่า ในการเชื่อมต่อที่มีความกว้างช่องสัญญาณสูงเวลาทั้งหมดที่ใช้ในการแสดงข้อมูลแบบสมบูรณ์จะน้อยกว่าเวลาทั้งหมดที่ใช้ในการแสดงตัวอย่างข้อมูล เนื่องจากเวลาในการแสดงตัวอย่างมีค่ามากกว่าเวลาที่ใช้ในการส่งข้อมูล และสำหรับการเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำเวลาทั้งหมดที่ใช้ในการแสดงตัวอย่างข้อมูลจะน้อยกว่าเวลาทั้งหมดที่ใช้ในการแสดงข้อมูลแบบสมบูรณ์ เนื่องจากการแสดงตัวอย่างจะทำให้ข้อมูลมีขนาดเล็กลงสามารถส่งข้อมูลไปให้แก่ไคลเอนต์โดยใช้เวลาน้อยลง สำหรับข้อมูลเวลาที่ใช้ในการแสดงข้อมูลภาพเคลื่อนไหวขนาดใหญ่มากแบบสมบูรณ์บนการเชื่อมต่อที่มีความกว้างช่องสัญญาณ 33.6 กิโลบิตต่อวินาที ในตารางที่ 4.2 ไม่ได้แสดงไว้เนื่องจากใช้เวลาในการทำงานนานมากเกินไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5
สรุปผลการวิจัย อภิปรายผล ปัญหาและข้อเสนอแนะ

เนื้อหาในบทสุดท้ายนี้จะได้กล่าวถึงส่วนสรุปผลการวิจัย อภิปรายผลการวิจัยที่ได้นำเสนอมาทั้งหมด ปัญหาในการวิจัย และข้อเสนอแนะสำหรับผู้สนใจ

5.1. **สรุปผลการวิจัย**

จากการศึกษาเพื่อออกแบบ และพัฒนาระบบแสดงตัวอย่างข้อมูลจากเว็บเพจสำหรับการเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำนั้น สามารถสรุปผลการวิจัยได้ดังนี้

1. การตรวจสอบความกว้างช่องสัญญาณที่เหมาะสม คือ การใช้แอปเพล็ต
2. การเก็บข้อมูลความกว้างช่องสัญญาณที่เหมาะสม คือ การเก็บในหน่วยความจำของเว็บเซิร์ฟเวอร์ และใช้วิธีการ Round – trip ข้อมูลโดยกำหนดจำนวนข้อมูลที่เก็บได้สูงสุด
3. ค่าความล่าช้าที่ดี คือ ค่าที่น้อย
4. การแสดงตัวอย่างข้อมูลภาพที่เหมาะสม คือ การตัดข้อมูลภาพให้เหลือพื้นที่ 25% แล้วแปลงไฟล์ข้อมูลภาพให้อยู่ในรูปแบบไฟล์เจเพ็กที่มีคุณภาพระดับ 5
5. การแสดงตัวอย่างข้อมูลภาพเคลื่อนไหวที่เหมาะสม คือ การตัดช่วงของภาพเคลื่อนไหวมา 25% แล้วลดขนาดของกรอบภาพเคลื่อนไหวให้เหลือ 160 x 128 จุด
6. การพัฒนาบริการแสดงตัวอย่างข้อมูลสำหรับอาปาเช่บนยูนิกซ์แพลตฟอร์มที่เหมาะสม คือ การใช้ Apache Perl Module

5.2. **อภิปรายผลการวิจัย**

ผลการวิจัยที่ได้มีข้อสังเกตในประเด็นต่าง ๆ ดังนี้

1. การมีความกว้างช่องสัญญาณสูงไม่ได้หมายความว่า จะสามารถรับส่งข้อมูลได้เร็วกว่าการเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำกว่า เนื่องจากความเร็วในการรับส่งข้อมูลขึ้นกับปัจจัยอื่น ๆ ด้วย เช่น ปริมาณการใช้งานเน็ตเวิร์กในขณะนั้น เป็นต้น
2. การตรวจสอบความกว้างช่องสัญญาณโดยการ Ping มีขั้นตอนการทำงานน้อย ทำงานได้รวดเร็ว แต่ไม่สามารถใช้งานได้กับไคลเอนต์ที่มีการติดตั้งไฟล์วอล์ที่ไม่อนุญาตให้โพรโทคอล ICMP ผ่านเข้าออกได้
3. การตรวจสอบความกว้างช่องสัญญาณโดยการใช้แอปเพล็ต สามารถทำงานได้กับไคลเอนต์ที่มีการติดตั้งไฟล์วอล์ที่ไม่อนุญาตให้โพรโทคอล ICMP ผ่านเข้าออกได้ แต่มีขั้นตอนการทำงานมาก ทำงานได้ช้า และโปรแกรมบราวเซอร์ที่ไคลเอนต์ใช้จะต้องมีจาวาอินเทอร์พรีเตอร์ ติดตั้งอยู่ด้วย

4. การกำหนดจำนวนข้อมูลที่จัดเก็บในหน่วยความจำของเว็บเซิร์ฟเวอร์ต้องคำนึงทรัพยากรของเว็บเซิร์ฟเวอร์ด้วย การกำหนดค่าที่สูงหรือต่ำเกินไปอาจทำให้เว็บเซิร์ฟเวอร์เกิดปัญหาในการทำงานได้

5.3. ปัญหาในการวิจัย

ผู้วิจัยได้พบปัญหาในการทำงานวิจัยอยู่ 2 ประการหลัก ๆ คือ

1. การตรวจสอบความกว้างช่องสัญญาณ เนื่องจากการตรวจสอบความกว้างช่องสัญญาณทำในระหว่างขั้นตอนการตอบสนองต่อคำร้องขอจากไคลเอนต์ ซึ่งเกิดขึ้นภายหลังการสร้างทีซีพีคอนเน็กชั่นระหว่างไคลเอนต์ และเว็บเซิร์ฟเวอร์แล้ว ทำให้มีการตรวจสอบความกว้างช่องสัญญาณทุกคำร้องขอแม้ว่าจะเป็นคำร้องขอจากไคลเอนต์เดิมก็ตาม การแก้ปัญหาโดยการเก็บข้อมูลความกว้างช่องสัญญาณสามารถลดความถี่ในการตรวจสอบลงได้บ้าง แต่เป็นการเพิ่มการทำงานให้กับเว็บเซิร์ฟเวอร์แทน
2. โปรแกรมที่ใช้สำหรับแสดงตัวอย่างข้อมูลภาพ และภาพเคลื่อนไหว มีให้เลือกใช้น้อย และบางโปรแกรมไม่มีอินเตอร์เฟสสำหรับภาษาต่าง ๆ ทำให้การใช้งานจำเป็นต้องรันจาก Shell ซึ่งทำให้เกิด Overhead ในการทำงานเพิ่มขึ้น

5.4. ข้อเสนอแนะ

จากผลการศึกษาครั้งนี้พบว่ามีข้อเสนอแนะสำหรับระบบแสดงตัวอย่างข้อมูลจากเว็บเพจสำหรับการเชื่อมต่อที่มีความกว้างช่องสัญญาณต่ำ ดังนี้

1. เพิ่มปัจจัยเวลาในการลบข้อมูลความกว้างช่องสัญญาณที่จัดเก็บไว้ในหน่วยความจำของเว็บเซิร์ฟเวอร์ เพื่อให้การตรวจสอบความกว้างช่องสัญญาณมีความเที่ยงตรงมากยิ่งขึ้น
2. ปรับปรุงการทำงานของการแสดงตัวอย่างข้อมูลให้มีการทำแคชเพื่อลดเวลาในการทำงานที่ซ้ำซ้อนกันลง
3. ปรับปรุงการทำงานของระบบให้ไคลเอนต์สามารถระบุค่าความล่าช้า และรูปแบบในการแสดงตัวอย่างข้อมูลของตัวเองได้
4. ทำการวิเคราะห์คุณภาพของตัวอย่างของข้อมูลที่ใช้วิธีการแสดงตัวอย่างแบบต่าง ๆ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

[1] Balachander Krishnamurthy, and Jennifer Rexford, 2001., *Web Protocols and Practice: HTTP/1.1, Networking Protocols, Caching, and Traffic Measurement*, (672 p.) Addison-Wesley Publishing Company.

[2] Ben Laurie, and Peter Laurie, 1999., *Apache: The Definitive Guide*, Second Edition, (385 p.) O'Reilly & Associates.

[3] Lincoln Stein, and Doug MacEachern, 1999., *Writing Apache Modules with Perl and C*, (724 p.) O'Reilly & Associates.

[4] Larry Wall, Tom Christiansen, and Jon Orwant, 2000., *Programming Perl*, Third Edition, (1092 p.) O'Reilly & Associates.

[5] H. M. Deitel, and P. J. Deitel, 1999., *Java : how to program*, Third Edition, (1597 p.) Prentice-Hall.

[6] วีระศักดิ์ ซึงถาวร, 2545., *Java Programming Volume II*., กรุงเทพฯ, (432 หน้า) ซีเอ็ดยูเคชั่น.

[7] JPEG image compression FAQ, part 1/2, 2003., *[5] What are good quality settings for JPEG?*, [On-line]. Available from http://www.faqs.org/faqs/jpeg-faq/part1/section-5.html. Visited: 9/5/2001. 10:00.

[8] *ImageMagick – Image Conversion, Editing, and Composition*, [On-line]. Available from http://www.imagemagick.org/. Visited: 31/1/2001. 14:00.

[9] Laurent Alacoque, *mpgtx a command line MPEG audio/video/system file toolbox*, [On-ilne]. Available from http://mpgtx.sourceforge.net/. Visited: 15/3/2002. 13:00.

[10] Fabrice Bellard, *FFmpeg Multimedia System*, [On-line]. Available from http://ffmpeg.sourceforge.net/. Visited: 24/1/2003. 11:00.

[11] Rob Brown, Net::*Ping – check a remote host for reachability*, [On-line]. Available from http://search.cpan.org/author/BBB/Net-Ping-2.28/lib/Net/Ping.pm. Visited: 29/1/2003. 15:00.

[12] Colin McMillen, *Net::Ping::External - Cross-platform interface to ICMP "ping" utilities*, [On-line]. Available from http://search.cpan.org/author/COLINM/Net-Ping-External-0.11/External.pm. Visited: 29/1/2003. 15:00.

[13] Benjamin Sugars, *IPC::Shareable - share Perl variables between processes*, [On-line]. Available from http://search.cpan.org/author/BSUGARS/IPC-Shareable-0.60/lib/IPC/Shareable.pm. Visited: 29/1/2003. 16:00.

[14]. Catherine Plaisant, Gary Marchionini, and Anita Komlodi, 1997., *Bringing Treasures to the Surface: Previews and Overviews in a Prototype for the Library of Congress National Digital Library*, in Proc. of CHI 97, ACM New York.

[15]. Gary Marchionini, Catherine Plaisant, Tom Bruns et al., 1997., *User Interface for the Library of Congress National Digital Library*, [On-line]. Available from http://www.cs.umd.edu/hcil/ndl/. Visited: 12/12/2000. 21:00.

[16] Antonio Ortega, Fabio Carignano, Serge Ayer, and Martin Vetterli, 1997., *Soft Caching: Web Cache Management Techniques for Images*, in Proc. of IEEE Signal Processing Society 1997 Workshop on Multimedia Signal Processing, Princeton, New Jersey, USA.

[17] Jussi Kangasharju, Young Gap Kwon, and Antonio Ortega, *Design and Implementation of a Soft Caching Proxy*, Integrated Media Systems Center, Department of EE-Systems, University of Southern California, Los Angeles, CA.

[18] Martin F. Arlitt, and Carey L. Williamson, 1996., *Web Server Workload Characterization: The Search for Invariants (Extended Version)*, Department of Computer Science, University of Saskatchewan, SK, Cannada.

[19] Allison Woodruff, Paul M. Aoki, Eric Brewer, Paul Gauthier, and Lawrence A. Rowe, *An Investigation of Documents from the World Wide Web*, The Fifth International World Wide Web Conference, Paris, France, May 6 – 10, 1996.

[20] นพรัตน์ พรมมหาไชย, 2543., *การวิเคราะห์ความสัมพันธ์ระหว่างลักษณะของยูอาร์แอลกับการใช้งานบริการเว็บแคช*, วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียนวิทยานิพนธ์

นายนิทัศน์ สุทวีปราโมชานนท์ เกิดเมื่อวันที่ 11 มีนาคม พ.ศ.2520 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิทยาศาสตร์บัณฑิต ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2541 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตร์มหาบัณฑิต ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ.2542

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย