การปรับแต่งค่าของพร็อกซีเซิร์ฟเวอร์สำหรับการใช้งานโพรโตคอล HTTP และ FTP

PROXY SERVER TUNING FOR HTTP AND FTP

วิฑิต อินทรหะ
WITHIT INTRAHA

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2544

ISBN 974-648-248-3

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การปรับแต่งค่าของพร็อกซีเซิร์ฟเวอร์สำหรับการใช้งานโพรโตคอล HTTP และ FTP

PROXY SERVER TUNING FOR HTTP AND FTP

T040006

วิฑิต อินทรหะ

WITHIT INTRAHA

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พ.ศ. 2544

ISBN 974-648-248-3

เลขหมู่..................
เลขทะเบียน 40006
วัน, เดือน, ปี 20 ก.ค. 2544

.b..................
.i..................

PROXY SERVER TUNING FOR HTTP AND FTP

WITHIT INTRAHA

A THESIS SUBMITED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF SCIENCE IN COMPUTER SCIENCE

AND INFORMATION TECHNOLOGY

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

2001

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ISBN 974-648-248-3

COPYRIGHT 2001
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่ใช้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การปรับแต่งค่าของพร็อกซี่เซิร์ฟเวอร์สำหรับการใช้งานโพรโตคอล HTTP และ FTP

PROXY SERVER TUNING FOR HTTP AND FTP

ชื่อนักศึกษา นายวิฑิต อินทรหะ

รหัสประจำตัว 37064429

ปริญญา วิทยาศาสตรมหาบัณฑิต

สาขาวิชา วิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ

อาจารย์ผู้ควบคุมวิทยานิพนธ์ อาจารย์อัครินทร์ คุณกิตติ

คณะกรรมการสอบวิทยานิพนธ์	ลายมือชื่อ
อาจารย์อัครินทร์ คุณกิตติ	
รศ.ดร.รัตติกร วรากูลศิริพันธ์	
รศ.ดร.วิเชียร เปรมชัยสวัสดิ์	
ดร.ประจวบ วานิชชัชวาล	
ดร.โชติพัชร์ ภรณวลัย	

วัน/เดือน/ปี 25 เมษายน 2544 เวลา 11.00 น. เป็นต้นไป

สถานที่สอบ ณ ห้อง LAB 316 ชั้น 3 คณะเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัยรับรองแล้ว

(รศ.ดร.บุญวัฒน์ อัตชู)

คณบดีบัณฑิตวิทยาลัย

วันที่...31...**เดือน**...พฤษภาคม...**พ.ศ.**...2544...

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การปรับแต่งค่าของพร็อกซีเซิร์ฟเวอร์สำหรับการใช้งานโพรโตคอล HTTP และ FTP
นักศึกษา	นาย วิฑิต อินทรหะ
รหัสประจำตัว	37064429
ปริญญา	วิทยาศาสตร์มหาบัณฑิต
สาขาวิชา	วิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ
พ.ศ.	2544
อ.ผู้ควบคุมวิทยานิพนธ์	อาจารย์ อัครินทร์ คุณกิตติ

บทคัดย่อ

ในระบบเครือข่ายขนาดใหญ่พร็อกซีเซิร์ฟเวอร์มีความสำคัญที่ช่วยลดความซ้ำซ้อนและเวลาที่ต้องใช้ในการค้นหาข้อมูลจากเครือข่ายอินเตอร์เนท แต่ยังไม่มีวิธีการปรับตั้งค่าพารามิเตอร์ที่สอดคล้องกับพฤติกรรมการใช้งานของผู้ใช้ งานวิจัยนี้จึงได้นำเสนอวิธีการในการคำนวณหาค่าพารามิเตอร์ขนาดของดิสก์แคชและอายุอ้างอิงที่ต้องใช้ในการปรับตั้งพร็อกซีเซิร์ฟเวอร์ดังกล่าว โดยทำการศึกษาพารามิเตอร์จากล็อกไฟล์ซึ่งเป็นผลจากการทำงานของ Squid ได้แก่ ขนาดและอายุของออบเจกต์ที่ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์ โดยใช้หลักในการจัดประเภทของการร้องขอที่บ่งบอกถึงการเปลี่ยนช่วงอายุโดยนำค่าเวลาที่เกิดการร้องขอนั้นมาคำนวณอายุของแต่ละออบเจกต์แล้ว หาค่าเฉลี่ยเพื่อที่จะนำไปใช้เป็นค่าอายุอ้างอิงในการปรับตั้งพารามิเตอร์ reference_age ใน Squid จากนั้นได้ทำการเสนอแนะวิธีการคำนวณหาขนาดของดิสก์แคชที่จะนำไปปรับตั้งด้วยสมมุติฐานที่จะพยายามหาค่าขนาดของดิสก์ที่ทำให้อัตราการพบข้อมูลมีค่าสูงที่สุดโดยที่ยังคงทำให้ค่าอัตราส่วนของอัตราการพบข้อมูลต่อขนาดของดิสก์มีค่ามากที่สุด โดยใช้ข้อมูล อัตราการพบข้อมูลและขนาดของดิสก์แคช ในช่วงเวลาก่อนหน้ามาป้อนกลับเพื่อใช้ในการคำนวณหาขนาดของดิสก์แคชที่จะนำไปใช้ปรับตั้งค่า cache_dir สำหรับช่วงเวลาถัดไป โดยทำการทดลองปรับกับพร็อกซีเซิร์ฟเวอร์ของ สจล. เพื่อทดลองปรับค่าโดยใช้ข้อมูลจากการทำงานจริง ซึ่งสามารถหาค่าขนาดของดิสก์และอายุอ้างอิงได้ตามอัลกอริธึมข้างต้น โดยค่าอายุอ้างอิงที่ได้จากค่าเฉลี่ยอายุของออบเจกต์นั้นเป็นค่าที่เหมาะสมที่จะใช้ปรับค่า reference_age ใน Squid ในขณะที่เราสามารถประมาณค่าขนาดของดิลก์แคชที่ใกล้เคียงกับลักษณะการใช้งานจริงของผู้ใช้ได้ และยังสามารถนำอัลกอริธึมดังกล่าวไปใช้ในการปรับตั้งพร็อกซีเซิร์ฟเวอร์อื่นได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Proxy Server Tuning for HTTP and FTP
Student	Mr. Withit Intraha
Student ID.	37064429
Degree	Master of Science
Programme	Computer Science and Information Technology
Year	2001
Thesis Advisor	Mr. Akharin Khunkitti

ABSTRACT

In the enterprise network systemS, proxy server plays a significant role to reduce the repetition and time in retrieving data from the Internet. However, despite its importance, there has not been any study, which identifies methods of setting the parameter to correspond with user-behavior. This research proposes algorithms to calculate the reference age and disk cache size, which are the Squid's parameters obtained from Squid's log file. The calculation of the reference age is done by deploying the analysis of object's requests, which specify life change from Squid's log. Each object life can be calculated from the different time stamp of each life change request. Then, we use an average of the object life as the reference_age. Another is to present an algorithm calculating disk cache size which tries to determine the cache size maximizing both %hit rate and ratio of %hit rate and disk cache size. Collected data, which are actual disk cache size and %hit rate, are used to calculate a new disk cache size. Then, we use this as parameter cache_dir. Finally, the algorithms have been deployed on KMITL's proxy servers showing that our algorithms worked in real environment. It shows that the average object life can be used as an exact parameter reference_age. The calculated disk cache size algorithm can monitor and estimate disk cache size, which is derived from real user-behavior. Further more, these algorithms can be applied to other proxy servers too.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดี เพราะได้รับความเมตตากรุณา คำแนะนำและคำปรึกษาจาก อาจารย์ อัครินทร์ คุณกิตติ ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ ผู้วิจัยรู้สึกซาบซึ้งในความอนุเคราะห์จากท่าน และขอกราบขอบพระคุณเป็นอย่างสูง

กราบขอพระคุณบิดา มารดา และขอขอบคุณครอบครัวของผู้วิจัย ที่ได้ให้การสนับสนุนในการศึกษาและทำงานวิจัยทั้งทางด้านกำลังใจ และการเงิน

ขอขอบพระคุณอาจารย์ประจำคณะเทคโน โลยีสารสนเทศ เป็นอย่างสูงที่เสียสละเวลาอันมีค่าให้ข้อคิดแนวทางในบางจุดที่ผู้วิจัยติดปัญหาบางอย่าง ซึ่งมีส่วนช่วยทำให้ผู้วิจัยเข้าใจในปัญหานั้น

ขอขอบคุณ คุณนรฤทธิ์ สุนทรศารทูล หัวหน้าฝ่ายระบบเครือข่ายและการสื่อสารข้อมูล สำนักวิจัยและบริการคอมพิวเตอร์เป็นอย่างสูงที่เอื้อเฟื้อสถานที่และเครื่องพร็อกซีเซิร์ฟเวอร์ตลอดจนทรัพยากรที่ใช้ในงานวิจัยนี้ ตลอดจนเจ้าหน้าที่และผู้เกี่ยวข้องที่ให้การสนับสนุนและดูแลเป็นอย่างดี

ขอขอบคุณเจ้าหน้าที่คณะเทคโน โลยีสารสนเทศ ซึ่งให้ความอนุเคราะห์ในการอำนวยความสะดวกต่าง ๆ ในการทำงานตั้งแต่ต้น รวมถึงพี่ ๆ น้อง ๆ เพื่อน ๆ นักศึกษาทุกคนที่ช่วยเหลือให้คำแนะนำต่าง ๆ และให้กำลังใจต่อผู้วิจัยอย่างมาโดยตลอด

ขอบคุณดุษที่เป็นกำลังใจและช่วยเหลือ ในการแก้ปัญหาและอุปสรรคต่างๆ ด้วยดีตลอดมา

วิฑิต อินทรหะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญรูป	VII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 จุดมุ่งหมายและวัตถุประสงค์ของงานวิจัย	2
1.3 แนวคิดและหลักการที่ใช้ในงานวิจัย	2
1.4 ขอบเขตของงานวิจัย	2
1.5 ขั้นตอนในการดำเนินงานวิจัย	2
1.6 นิยามศัพท์และข้อตกลงเบื้องต้น	3
1.7 ประโยชน์ที่คาดว่าจะได้รับ	3
1.8 รายละเอียดของแต่ละบท	4
บทที่ 2 หลักการทำงานของพร็อกซีเซิร์ฟเวอร์	5
2.1 หลักการทำงานและองค์ประกอบที่เกี่ยวข้องเบื้องต้น	6
2.1.1 Squid พร็อกซีเซิร์ฟเวอร์ เวอร์ชั่น 2.3	6
2.1.2 ขั้นตอนการทำงานของ Squid พร็อกซีเซิร์ฟเวอร์	12
2.2 รูปแบบความสัมพันธ์ระหว่างพร็อกซีเซิร์ฟเวอร์	20
2.3 ไฟล์ Squid.conf กับพารามิเตอร์ที่มีผลต่อการทำงานของพร็อกซีเซิร์ฟเวอร์	25
2.4 ล็อกไฟล์	28
2.5 สรุป	33
บทที่ 3 การหาค่าพารามิเตอร์ที่สำคัญและเกี่ยวข้องกับการปรับแต่งพร็อกซีเซิร์ฟเวอร์	34
3.1 พารามิเตอร์ที่สำคัญและวัดได้จากล็อกไฟล์	34
3.2 อัลกอริธึมในการวัดค่าอายุและขนาดของออบเจกต์	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
3.3 การวิเคราะห์ความผิดพลาดที่เกิดจากอัลกอริธึมในการวัดอายุของออบเจกต์.....	39
3.3.1 การหาค่าความผิดพลาดที่เกิดจากอัลกอริธึมในการวัดอายุของออบเจกต์	39
3.3.2 การวิเคราะห์ผลกระทบของอัตราการร้องขอข้อมูล ที่มีผลต่อค่า .... ความผิดพลาดของอายุที่ได้จากอัลกอริธึมในการวัดอายุของออบเจกต์	40
3.4 การทดลองวัดค่าอายุและขนาดของออบเจกต์	42
3.5 สรุป	46
บทที่ 4 การปรับแต่งพร็อกซีเซิร์ฟเวอร์	47
4.1 พารามิเตอร์ที่เกี่ยวข้องกับการปรับแต่งพร็อกซีเซิร์ฟเวอร์	47
4.2 อัลกอริธึมในการปรับแต่งค่าดิสก์แคช	48
4.3 การทดลองปรับแต่งขนาดของดิสก์แคชและอายุอ้างอิง	53
4.3.1 การทดลองปรับค่าของดิสก์แคช	53
4.3.2 การทดลองปรับค่าอายุอ้างอิง	55
4.4 บทวิเคราะห์ผลการปรับแต่ง	59
บทที่ 5 สรุปผลงานวิจัยและข้อเสนอแนะ	61
5.1 สรุปผลงานวิจัยการปรับแต่งค่าขนาดของดิสก์แคชและอายุอ้างอิงสำหรับ .... Squid พร็อกซีเซิร์ฟเวอร์	61
5.2 ข้อเสนอแนะของการปรับแต่งค่าพร็อกซีเซิร์ฟเวอร์	63
เอกสารอ้างอิง	65
ภาคผนวก	67
บทความและผลงานวิจัยที่ได้รับการตีพิมพ์	67
ประวัติผู้เขียน	87

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

2.1 ฟิลด์ต่าง ๆ ของล็อกไฟล์ที่เกิดจากการประมวลผลการร้องขอของ Squid 28
3.1 ประเภทของการร้องขอที่มีผลต่อการหาอายุของออบเจกต์... 36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

2.1 โฟลว์ชาร์ตสรุปการทำงานเมื่อ Squid ได้รับการร้องขอจากไคลเอนท์ 14
2.2 ผลของการร้องขอที่พบออบเจกต์ในเมโมรีแคช.. 15
2.3.ผลของการร้องขอที่พบออบเจกต์ในเมโมรีแคชแต่หมดอายุ... 16
2.4 ผลของการร้องขอที่พบออบเจกต์ในดิสก์แคช 16
2.5 ผลของการร้องขอที่พบออบเจกต์ในดิสก์แคชแต่หมดอายุ.. 17
2.6 ผลของการร้องขอที่ไม่พบออบเจกต์ในแคชของ Squid 18
2.7 การจัดเก็บออกเจกต์ในแคชของ Squid 18
2.8 ระบบพร็อกซีเซิร์ฟเวอร์แบบ Single Proxy 21
2.9 ระบบพร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์แบบ Parent-Child... 22
2.10 ระบบพร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์แบบ Sibling 22
2.11 ตัวอย่างระบบพร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์แบบผสม.. 23
2.12 รายละเอียดรูปแบบของ ICP Message 24
3.1 หลักการหาอายุของออบเจกต์จากข้อมูลการร้องขอข้อมูลในพร็อกซีเซิร์ฟเวอร์ 35
3.2 ค่าความผิดพลาดของอายุของออบเจกต์ซึ่งถูกนอร์มอลไลซ์ (Normalized) ณ.อัตรา ....
การร้องขอต่าง ๆ 43
3.3 ค่าความผิดพลาดของอายุที่คำนวณซึ่งถูกนอร์มอลไลซ์ ณ. อัตราการร้องขอใน ....
1 ช่วงอายุต่าง ๆ 44
3.4 กราฟการกระจายของขนาดของออบเจกต์... 45
3.5 กราฟการกระจายของอายุของออบเจกต์ 45
4.1 ความสัมพันธ์ของอัตราการพบข้อมูลและขนาดของดิสก์แคช.. 48
4.2 ค่า R และ %Hit 53
4.3 ค่า R และ S ที่ใช้ในการทำดิสก์แคชจริง (S_real) 54
4.4 แนวโน้มของค่า R ที่ได้จากค่าเฉลี่ยของค่า R ในรอบ 1 วัน 54
4.5 ค่า S_tune, S_real และ S_cal ในช่วงชั่วโมงที่ 80 ถึง 140.. 55
4.6 ค่า R และ S เมื่อใช้ค่า refernce_age เท่ากับ 2 สัปดาห์ 56
4.7 ค่า R และ H เมื่อใช้ค่า refernce_age เท่ากับ 2 สัปดาห์.. 56
4.8 ค่า R และ S เมื่อใช้ค่า refernce_age เท่ากับ 6 ชั่วโมง 57
4.9 ค่า R และ H เมื่อใช้ค่า refernce_age เท่ากับ 6 ชั่วโมง... 57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.10 ค่า R และ S เมื่อใช้ค่า refernce_age เท่ากับ 1 ชั่วโมง 58

4.11 ค่า R และ H เมื่อใช้ค่า refernce_age เท่ากับ 1 ชั่วโมง 58

4.12 ค่า S และ R เมื่อใช้ค่า reference_age ต่าง ๆ 59

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันความต้องการในการเชื่อมต่อเข้าสู่เครือข่ายอินเทอร์เน็ตมีเพิ่มมากขึ้นอย่างรวดเร็วและไม่มีทีท่าว่าจะชลอตัวลง ทั้งนี้เนื่องจากในปัจจุบันเครือข่ายอินเทอร์เน็ตได้บรรจุไว้ซึ่งข้อมูลข่าวสาร และบริการ ที่หลากหลายมากกว่าในยุคแรกเริ่ม จึงเป็นเหตุจูงใจให้มีความพยายามติดต่อเข้าสู่เครือข่ายนี้เป็นจำนวนมาก

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง เป็นหน่วยงานสถาบันการศึกษาแห่งหนึ่งที่เล็งเห็นความสำคัญในการให้บริการเชื่อมต่อเข้าสู่เครือข่ายอินเทอร์เน็ตแก่ ผู้บริหาร ข้าราชการ เจ้าหน้าที่ และ นักศึกษา ของทางสถาบันฯ เพื่อใช้ประโยชน์ในการสืบค้นข้อมูล แต่เนื่องจากความต้องการในการเข้าสืบค้นข้อมูลจากอินเทอร์เน็ตของผู้ใช้ได้เพิ่มปริมาณขึ้นอย่างรวดเร็วตามที่ได้กล่าวข้างต้น ทำให้เส้นทางการเชื่อมต่อออกสู่อินเทอร์เน็ตของสถาบันฯ เกิดความคับคั่งของการจราจรของข้อมูลบนเครือข่ายคอมพิวเตอร์ ส่งผลให้เกิดความล่าช้าและเสียเวลาในการรอคอยข้อมูลที่ต้องการ เพื่อเป็นการลดความซ้ำซ้อนในการร้องขอข้อมูลจากเครือข่ายคอมพิวเตอร์ภายนอกสถาบันฯ จึงได้มีการนำพร็อกซีเซิร์ฟเวอร์เข้ามาเพื่อทำหน้าที่เป็นตัวแทนในการเรียกค้นข้อมูลจากภายนอกและทำสำเนาเก็บไว้เพื่อให้ผู้ใช้ที่ต้องการข้อมูลเดียวกันนั้นสามารถนำข้อมูลที่ทำสำเนาไว้ไปใช้ได้ทันทีอันจะส่งผลให้การใช้งานเส้นทางการเชื่อมต่อออกสู่อินเทอร์เน็ตเป็นไปอย่างมีประสิทธิภาพมากขึ้นอันจะทำให้สามารถประหยัดค่าใช้จ่ายในการเชื่อมต่อเข้าสู่อินเทอร์เน็ตได้ในระดับหนึ่ง

ด้วยเหตุนี้ประสิทธิภาพในการทำงานของพร็อกซีเซิร์ฟเวอร์จึงเป็นปัจจัยสำคัญที่จะสะท้อนถึงความสามารถในการให้บริการแก่ผู้ใช้งาน และช่วยประหยัดค่าใช้จ่ายในการเชื่อมต่อสู่เครือข่ายภายนอกอีกด้วย ปัญหาที่เกิดขึ้นคือการนำพร็อกซีเซิร์ฟเวอร์เข้ามาใช้ประโยชน์ในการประหยัดค่าใช้จ่ายในการเชื่อมต่อเข้าสู่อินเทอร์เน็ตนั้น เครื่องพร็อกซีเซิร์ฟเวอร์มีการปรับตั้งค่าพารามิเตอร์ต่าง ๆ ที่เหมาะสมแล้วหรือยัง รวมไปถึงความเหมาะสมของฮาร์ดแวร์ (เช่น ขนาดของเนื้อที่เก็บข้อมูลบนดิสก์) อีกด้วย

จากการสืบค้นงานวิจัยที่เกี่ยวข้องกับการปรับตั้งพร็อกซีเซิร์ฟเวอร์นั้นส่วนมากจะพบงานวิจัยที่เกี่ยวข้องกับการปรับตั้งอัลกอริธึมในการจัดเก็บข้อมูลในแคช [2] [9] [16] [19] [22] และความพยายามในการทำนายพฤติกรรมการร้องขอข้อมูลล่วงหน้า [8] [13] [14] [23] [24] แต่ในงานวิจัยนี้จะมุ่งความสนใจมาที่การเสนอแนวคิดและวิธีการในการปรับตั้งค่าขนาดของดิสก์แคชและอายุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อ้างอิงของออบเจกต์ที่จัดเก็บในดิสก์แคชโดยอาศัยข้อมูลพฤติกรรมของผู้ใช้ที่แฝงอยู่ในล็อกไฟล์ ซึ่งเป็นไฟล์ที่บันทึกผลการทำงานของเครื่องพร็อกซีเซิร์ฟเวอร์ในการให้บริการในแต่ละการร้องขอที่ผู้ใช้ร้องขอเข้ามา

1.2 จุดมุ่งหมายและวัตถุประสงค์ของงานวิจัย

1.ศึกษาหลักการทำงานตลอดจนองค์ประกอบแวดล้อมในการทำงานของพร็อกซีเซิร์ฟเวอร์

2.นำเสนอวิธีการคำนวณหาค่าขนาดของดิสก์แคชและอายุของออบเจกต์เพื่อนำไปใช้ปรับตั้งให้กับพร็อกซีเซิร์ฟเวอร์

1.3 แนวคิดและหลักการที่ใช้ในงานวิจัย

เนื่องจากเครื่องคอมพิวเตอร์ที่นำมาทำเป็นพร็อกซีเซิร์ฟเวอร์นั้นต้องเป็นเครื่องที่มีทรัพยากรที่สูงมาก ตัวอย่างเช่น ต้องมีหน่วยประมวลผลกลางความเร็วสูง เพื่อให้สามารถรองรับการประมวลผลการร้องขอที่จะมีเข้ามาเป็นจำนวนมากได้เป็นอย่างดี เพื่อให้สามารถงานวิจัยนี้เกี่ยวข้องกับการเรียกค้นข้อมูลผ่านเครือข่ายอินเทอร์เน็ต อีกทั้งต้องมีพื้นที่ของหน่วยความจำหลัก และพื้นที่จัดเก็บข้อมูลบนดิสก์ที่เพียงพอในการจัดเก็บข้อมูลที่ถูกร้องขอ ซึ่งส่งผลทำให้ราคาของเครื่องคอมพิวเตอร์ที่ต้องการสูงขึ้นตามความสามารถในการประมวลและและเนื้อที่เก็บข้อมูล

งานวิจัยนี้จึงมีแนวคิดในการที่จะศึกษาหลักการทำงานและองค์ประกอบที่เกี่ยวข้องในการทำงานของพร็อกซีเซิร์ฟเวอร์ เพื่อหาวิธีการในการปรับตั้งพร็อกซีเซิร์ฟเวอร์เพื่อหาจุดเหมาะสมระหว่างความสามารถในการให้บริการของเครื่องพร็อกซีเซิร์ฟเวอร์กับทรัพยากรของเครื่องที่ต้องใช้ไปเพื่อแลกกับประสิทธิภาพในการทำงาน

1.4 ขอบเขตของงานวิจัย

ในการศึกษาหลักการทำงานของพร็อกซีเซิร์ฟเวอร์นั้นจะทำการศึกษาโดยเน้นที่การทำงานของ Squid เพื่อที่จะศึกษารายละเอียดในขั้นตอนการทำงาน พารามิเตอร์ที่เกี่ยวข้อง และผลการทำงานที่แสดงผ่านทางล็อกไฟล์ ในส่วนของการปรับตั้งค่าเพื่อเพิ่มประสิทธิภาพของพร็อกซีเซิร์ฟเวอร์นั้นจะมุ่งความสนใจไปที่การปรับตั้งค่าพารามิเตอร์สำหรับพร็อกซีที่ทำงานเป็นเครื่องเดี่ยว ซึ่งจะทำการทดลองกับเครื่องพร็อกซีเซิร์ฟเวอร์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง โดยทำการศึกษาพารามิเตอร์ในส่วนของ เปอร์เซ็นต์การพบข้อมูล(%Hit) ขนาดพื้นที่ของดิสก์แคช (Disk Cache Size) อายุและขนาดของออบเจกต์ ที่คาดว่าจะมีผลต่อพร็อกซีเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ขั้นตอนในการดำเนินงานวิจัย

ขั้นตอนในการดำเนินงานประกอบด้วย ขั้นตอนหลักๆ ดังนี้

1 กำหนดหัวข้อ เป้าหมาย จุดประสงค์ และขอบเขตการทำวิทยานิพนธ์

2 ศึกษาทฤษฎีและหลักการพื้นฐานที่เกี่ยวข้องกับพร็อกซีเซิร์ฟเวอร์

3 ศึกษาและจัดหาอุปกรณ์เครื่องมือและวิธีการที่เหมาะสมที่จะใช้ในการวิจัย

4 ออกแบบการทดลอง

5 ทำการทดลองและวัดผล

6 ปรับปรุงแก้ไข

7 จัดทำเอกสารประกอบวิทยานิพนธ์

1.6 นิยามศัพท์และข้อตกลงเบื้องต้น

จากหัวข้องานวิจัย นั้น การปรับแต่งพร็อกซีเซิร์ฟเวอร์สำหรับการใช้งาน HTTP และ FTP นั้น ในส่วนของการใช้งาน โพรโตคอล FTP นั้นให้หมายความถึงการใช้โพรโตคอล FTP ผ่าน HTTP Client เท่านั้น ไม่ได้หมายความรวมไปถึงการใช้โพรโตคอล FTP ผ่าน FTP Client

เพื่อให้มีความเข้าใจที่ตรงกันและเพื่อป้องกันความสับสนในการใช้คำอธิบายในเนื้อหาของงานวิจัยฉบับนี้ จึงขอนิยามความหมายของคำศัพท์บางคำเพื่อใช้กล่าวถึงในลำดับต่อไปดังนี้

ออบเจ็กต์ หมายถึง ไฟล์หรือแฟ้มข้อมูลประเภทต่าง ๆ ที่ผู้ใช้งานร้องขอผ่านพร็อกซีเซิร์ฟเวอร์

Squid หมายถึง โปรแกรมที่ทำหน้าที่เป็นพร็อกซีเซิร์ฟเวอร์

ล็อกไฟล์ หมายถึง แฟ้มข้อมูลตัวอักษร (Text File) ที่บันทึกผลการทำงานของ Squid

พร็อกซีเซิร์ฟเวอร์ หมายถึง เว็บพร็อกซีเซิร์ฟเวอร์ที่ใช้โพรโตคอล HTTP ในการให้บริการเรียกค้นข้อมูลเป็นหลัก

ไคลเอนท์ หมายถึง แอพพลิเคชั่นประเภทเว็บไคลเอนท์ เช่น เว็บบราวเซอร์ และหมายความรวมถึงเครื่องพร็อกซีเซิร์ฟเวอร์อื่นที่ร้องขอออบเจกต์ด้วย

1.7 ประโยชน์ที่คาดว่าจะได้รับ

1.เข้าใจหลักการทำงานและทราบถึงพารามิเตอร์ที่สำคัญของพร็อกซีเซิร์ฟเวอร์ ตลอดจนสามารถตีความผลการทำงานได้จากล็อกไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

2.สามารถนำอัลกอริธึมในการวัดค่าพารามิเตอร์ที่สำคัญของพร็อกซีเซิร์ฟเวอร์ไปใช้งานจริงเพื่อศึกษาลักษณะของออบเจกต์ที่มีอยู่ในอินเทอร์เน็ต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. สามารถนำอัลกอริธึมในการปรับตั้งพร็อกซีเซิร์ฟเวอร์ไปใช้หาค่าที่เหมาะสมสำหรับพร็อกซีเซิร์ฟเวอร์เพื่อเพิ่มประสิทธิภาพในการใช้งานพร็อกซีเซิร์ฟเวอร์ได้

1.8 รายละเอียดของแต่ละบท

ในบทที่ 2 จะกล่าวถึงหลักการทำงานของพร็อกซีเซิร์ฟเวอร์ตลอดจนองค์ประกอบที่มีส่วนเกี่ยวข้องกับการทำงานของพร็อกซีเซิร์ฟเวอร์ พารามิเตอร์ที่สำคัญในการปรับแต่งค่าเริ่มต้นในการทำงาน รูปแบบความสัมพันธ์ในการเชื่อมต่อพร็อกซีเซิร์ฟเวอร์ และความหมายของล็อกไฟล์ของ Squid

ในบทที่ 3 จะนำเสนอวิธีการในการคำนวณหาพารามิเตอร์ที่สำคัญและเกี่ยวข้องกับการปรับแต่งพร็อกซีเซิร์ฟเวอร์ โดยจะกล่าวถึงอัลกอริธึมที่ใช้ในการวัดค่าอายุและขนาดของ ออบเจกต์ที่ถูกร้องขอจากอินเทอร์เน็ตผ่านพร็อกซีเซิร์ฟเวอร์รวมถึงการวิเคราะห์ความผิดพลาดที่เกิดจากอัลกอริธึมที่ใช้ในการวัดด้วย จากนั้นได้ทำการแสดงผลการทดลองเพื่อสนับสนุนยืนยันว่าขอบเขตของความผิดพลาดที่หาได้จากการทดลองนั้นมีค่าสอดคล้องกับที่ได้วิเคราะห์เอาไว้จากวิธีที่เสนอไป และสุดท้ายจะแสดงผลการทำลองในการใช้อัลกอริธึมนี้ในการทดลองวัดค่าอายุและขนาดของออบเจกต์จริงจากล็อกไฟล์ที่เกิดจากการทำงานของเครื่องพร็อกซีเซิร์ฟเวอร์

ในบทที่ 4 จะนำเสนอแนวคิดและวิธีการในการปรับแต่งพารามิเตอร์ต่าง ๆ ของพร็อกซีเซิร์ฟเวอร์เพื่อเพิ่มประสิทธิภาพในการใช้งานของเครื่องพร็อกซีเซิร์ฟเวอร์ จากนั้นจะแสดงผลของการปรับแต่งพร็อกซีเซิร์ฟเวอร์ตามวิธีการที่นำเสนอ พร้อมทั้งวิเคราะห์ผลการปรับแต่ง

ในบทที่ 5 จะเป็นบทสรุปผลงานวิจัยในการปรับแต่งพร็อกซีเซิร์ฟเวอร์ทั้งหมด และข้อเสนอแนะในการปรับแต่งค่าของพร็อกซีเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
หลักการทำงานของพร็อกซีเซิร์ฟเวอร์

ในบทนี้จะกล่าวถึงหลักการทำงานของพร็อกซีเซิร์ฟเวอร์โดยทั่วไปและกล่าวถึงพร็อกซีเซิร์ฟเวอร์ที่ได้มีการพัฒนาขึ้นมาใช้งาน จากนั้นจะกล่าวถึง Squid ซึ่งเป็นพร็อกซีเซิร์ฟเวอร์ที่สนใจจะศึกษาโดยจะอธิบายถึงส่วนประกอบต่าง ๆ หลักการทำงาน พารามิเตอร์ที่สำคัญในการปรับตั้งการทำงานของ Squid และ รูปแบบของล็อกไฟล์และพารามิเตอร์ที่มีอยู่ในล็อกไฟล์เพื่อให้เข้าใจและทราบถึงภาพรวมในการทำงานของพร็อกซีเซิร์ฟเวอร์

2.1 หลักการทำงานและองค์ประกอบที่เกี่ยวข้อง

การทำงานของพร็อกซีเซิร์ฟเวอร์จะใช้หลักการของไคลเอนท์/เซิร์ฟเวอร์ที่มีตัวเครื่องพร็อกซีเซิร์ฟเวอร์เป็นผู้ให้บริการแก่ไคลเอนท์ซึ่งก็คือ HTTP Client หรือ FTP Client โดยบริการในที่นี้ก็คือการเป็นตัวแทนของไคลเอนท์เหล่านั้นในการไปดึงข้อมูลจากภายนอกมาให้แก่ไคลเอนท์ตามที่ได้รับการร้องขอมา ทั้งนี้พร็อกซีเซิร์ฟเวอร์ยังสามารถทำการจัดเก็บข้อมูลเหล่านั้นไว้ชั่วระยะเวลาหนึ่ง (การทำแคช) ซึ่งถ้ามีการร้องขอข้อมูลเดียวกันเข้ามาพร็อกซีเซิร์ฟเวอร์ก็สามารถนำข้อมูลที่จัดเก็บไว้ในแคชส่งให้ไคลเอนท์ได้เลยโดยไม่ต้องไปดึงมาจากภายนอกอีก

แอพพลิเคชั่นเว็บแคช ที่เป็นที่นิยมในปัจจุบันมีอยู่อย่างหลากหลายทั้งที่พัฒนาขึ้นเพื่อจุดประสงค์ทางการค้าและไม่หวังผลตอบแทน โดยแอพพลิเคชั่นจะมีความสามารถในการทำงานที่แตกต่างกันออกไป ในที่นี้จะขอแบ่งแอพพลิเคชั่นออกตามกลุ่มของผู้พัฒนา [6]

CERN/W3C เป็นต้นฉบับของพร็อกซีแคช โดยพัฒนาเริ่มแรกที่ CERN ต่อมาถูก W3C นำมาพัฒนาต่อ

Harvest Cache เป็นส่วนประกอบหนึ่งของ Harvest Project คือออบเจกต์แคชและที่นี่เป็นแห่งแรกที่มีการพัฒนาโพรโตคอล ICP (Internet Cache Protocol) โดยตัว Harvest Cache ถูกนำไปพัฒนาต่อโดย SQUID และ Netscape โดยสามารถรองรับการทำงานของโพรโตคอล: HTTP, FTP, gopher, ICP

Netscape Proxy Server: ถูกพัฒนาโดยนักพัฒนาคนสำคัญที่เป็นคนสร้าง CERN Proxy โดยสามารถรองรับการทำงานของโพรโตคอลได้หลายตัว เช่น HTTP, FTP, Gopher, SSL, ICP. CARP (Cache Array Routing Protocol)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DeleGate เป็นพร็อกซีเซิร์ฟเวอร์ที่ทำงานได้บนหลากหลาย Platform ของระบบปฏิบัติการ ไม่ว่าจะเป็น Unix, Windows, OS/2 และสามารถรองรับโพรโตคอลเช่น HTTP,

FTP, Gopher, NNTP, POP, SMTP, Telnet, WAIS, X, CU-SeeMe, Socks, SSL และ ICP version 2

Microsoft proxy: ทำงานภายใต้สภาพแวดล้อมของ Windows NT และทำงานร่วมกับโพรโตคอล HTTP และ CARP

Wcol/Catalyst: อาศัยการทำ perfecting เพื่อลดความล่าช้าและเพิ่ม bandwidth

Novell BorderManager FastCache: รองรับโพรโตคอล: HTTP, ICP ได้

Apache: เป็นเว็บเซิร์ฟเวอร์ที่มีความสามารถในการทำแคชด้วย

Cacheflow: ใช้เทคนิค Prefecting ในการลดเวลาแฝงในการร้องขอข้อมูล

MOWS: ใช้ภาษาจาวาในการพัฒนา สามารถทำงานร่วมกับ โพรโตคอล ICP ได้

นอกจากนี้ยังมีแอพพลิเคชั่นเว็บแคชบ้างตัวซึ่งไม่ได้กล่าวถึงเนื่องจากในงานวิจัยนี้มุ่งเน้นศึกษาการทำงานของ Squid เป็นหลัก เนื่องจากได้ถูกนำมาใช้ในเครื่องพร็อกซีเซิร์ฟเวอร์ของทางสถาบันเทคโนโลยีพระจอมเกล้าฯ ลาดกระบัง

2.1.1 Squid พร็อกซีเซิร์ฟเวอร์ เวอร์ชั่น 2.3

Squid คือ Internet Proxy Caching application ซึ่งมีจุดเริ่มต้นมาจากการพัฒนาโปรแกรมของ Harvest Project ซึ่งไม่ได้มีจุดมุ่งหมายหลักในการพัฒนาแคช โดยได้รับเงินทุนสนับสนุนงานวิจัยจาก National Laboratory of Network Research (NLANR) Squid เป็นแอพพลิเคชั่นที่เผยแพร่ Source Code โดยไม่คิดค่าใช้จ่าย จึงมีผู้นำ Squid ไปพัฒนาเพิ่มความสามารถอื่น ๆ และแก้ไขข้อผิดพลาดที่เกิดขึ้นอย่างแพร่หลาย Squid จึงเป็นที่นิยมเพราะผู้ที่มีความรู้ทางด้านการเขียนโปรแกรมสามารถจะสามารถปรับปรุงประสิทธิภาพของมันตามความต้องการได้โดย Squid มีความสามารถในการทำ Proxying และ Caching โดยใช้โพรโตคอล HTTP, FTP และ URL แบบอื่น ๆ เช่น SSL และสามารถจะทำงานแบบ Cache Hierarchies โดยใช้โพรโตคอล ICP, HTCP, CRAP, Cache-digest [3], [17] ในการสื่อสารระหว่างกัน การที่แคชจะสามารถทำงานได้อย่างมีประสิทธิภาพได้นั้น จะต้องประกอบด้วยประสิทธิภาพที่ดีของระบบโดยรวม ซึ่งองค์ประกอบที่สำคัญต่อประสิทธิภาพของแคช คือ เวลาที่ใช้ในการเข้าถึงข้อมูลในดิสก์ หน่วยความจำทั้งหมดของระบบ Disk-Throughput และความสามารถในการประมวลผลของ CPU

การที่จะทำให้แคชมีเสถียรภาพนั้นผู้ดูแลระบบควรจะมีการเตรียมพร้อมเพื่อรับมือกับปัญหาที่จะเกิดขึ้นไม่ว่าจะเป็นการเตรียมอะไหล่ที่จำเป็นในกรณีฉุกเฉินหรือการเตรียมระบบสำรองที่ติดตั้งระบบปฏิบัติการและ Squid เอาไว้เมื่อเกิดปัญหากับเครื่องแคชหลักก็จะสามารถนำเครื่องสำรองมาใช้งานแทนที่ได้ทันที การตัดสินใจเลือกระบบที่จะนำมาใช้ร่วมกับ Squid นั้นมีองค์ประกอบหลาย ๆ อย่างที่ต้องคำนึง ไม่ว่าจะเป็นค่าปริมาณสูงสุดของการร้องขอต่อนาที (Peak number of Request per minute) ซึ่งจะแสดงให้เห็นจำนวนของออบเจกต์ที่ถูกร้องขอจากไคลเอนท์

และสามารถนำมาคำนวณหาภาระงานที่แคชจะต้องแบกรับขณะทำงาน เป็นการยากที่จะคำนวณค่าการร้องขอสูงสุดนี้ ขึ้นอยู่กับลักษณะพฤติกรรมการใช้งานอินเทอร์เนตของผู้ใช้ เช่นเดียวกับการตัดสินใจเลือก ฮาร์ดแวร์ ที่เหมาะสม หากไม่มีสถิติการใช้งานอินเทอร์เนต เราอาจจะติดตั้ง แคชทดสอบเพื่อประเมิน การร้องขอ ที่จะเกิดขึ้นก็ได้ จะเป็นการดี หากเราประเมินฮาร์ดแวร์ ที่เราต้องการมากกว่า ความต้องการใช้งานในปัจจุบัน เพื่อให้สามารถรองรับความต้องการที่จะเพิ่มขึ้นตามมาในอนาคต

Squid สามารถทำงานร่วมกับ ระบบปฏิบัติการ Unix ได้เกือบทุก Platform เนื่องจาก Squid เขียนบน Digital Unix ที่ทำงานกับ GNU C compiler จึงสามารถนำ Squid ไปติดตั้งบน OS ที่มี compiler ดังกล่าว โดยการติดตั้งจะเลือกนำ Source code มา compile หรือนำ Binary Version มาติดตั้ง ซึ่งสามารถกระทำได้ง่ายกว่า แต่อาจจะมีปัญหาเรื่องความปลอดภัยที่เกิดจากการทำงานของโปรแกรมย่อยต่าง ๆ ที่แอบแฝงอยู่ Squid โดยผู้ไม่หวังดีได้

ส่วนประกอบของ Squid มีส่วนประกอบที่สำคัญโดยทั่วไปดังนี้ [5]

1. Client side: เป็นส่วนที่ใช้รับการเชื่อมต่อจาก ไคลเอนท์ แล้วทำการตีความการร้องขอที่ได้รับ และประมวลผลการร้องขอนั้น ๆ ผลของการประมวลผลตรงส่วนนี้ก็จะระบุผลของการร้องขอที่มีเข้ามาด้วยว่ามีผลการทำงานเป็นแบบใด เช่น HIT MISS หรือ REFRESH เป็นต้น ซึ่งจะมีการเก็บข้อมูลของสถานะของแต่ละการร้องขอที่ประมวลผลเสร็จสิ้นไว้เพื่อใช้งานต่อไป เช่นการเก็บบันทึกลงล็อกไฟล์ (Log File)
2. Server side: มีหน้าที่ส่งผ่านการร้องขอจากไคลเอนท์ที่มีผลของการร้องขอว่าไม่พบข้อมูลในแคช (Cache MISS) ไปยังเครื่องเซิร์ฟเวอร์อื่น โดยขึ้นอยู่กับประเภทของโพรโตคอลที่ไคลเอนท์ร้องขอเข้ามาอีกด้วย การส่งผ่านการร้องขอนี้อาจจะส่งผ่านไปยังเว็บเซิร์ฟเวอร์ต้นทางที่ไคลเอนท์ต้องการร้องขอข้อมูลหรืออาจจะเป็นเครื่องพร็อกซีเซิร์ฟเวอร์อื่น ๆ ก็ได้ ทั้งนี้แล้วแต่การปรับตั้งค่าของพร็อกซีเซิร์ฟเวอร์ว่าให้มีการทำงานร่วมกับพร็อกซีเซิร์ฟเวอร์เครื่องอื่น ๆ ด้วยหรือไม่อย่างไร โดยที่ทุก ๆ การร้องขอจากฝั่งเซิร์ฟเวอร์ไปยังเครื่องเซิร์ฟเวอร์อื่น ๆ ดังกล่าวมาแล้วนั้นจะเป็นการส่งผ่านในรูปแบบของโพรโตคอล HTTP เท่านั้น แม้ว่าการร้องขอเริ่มต้นจากทางไคลเอนท์นั้นจะเป็นการร้องขอด้วยโพรโตตอล FTP หรือ Gopher ก็ตาม โพรโตคอล WAIS และ Gopher นั้นไม่ค่อยได้รับความสนใจมากนักเนื่องจากในปัจจุบันมันมีสัดส่วนของปริมาณของข้อมูลและการร้องขอผ่านอินเทอร์เนทที่น้อยมากเมื่อเทียบกับ HTTP

เอกสารนี้เป็นเอกสารที่สงวน ... การค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้าม ...

3. Storage Manager: ส่วนนี้จะเป็นส่วนเชื่อมประสานระหว่าง client side และ server side โดยจะทำหน้าที่จัดการกับออบเจกต์ที่เก็บอยู่ในแคช ทั้งการค้นหาออบเจกต์จากแคชเมื่อถูกร้องขอ และการจัดการแทนที่ออบเจกต์เก่าด้วยออบเจกต์ที่ใหม่กว่าในกรณีที่ไม่มีเนื้อที่ดิสก์เหลือพอที่จะเก็บออบเจกต์ใหม่ โดยอาศัย วิธีการในการแทนที่ (Replacement Algorithm) แบบต่าง ๆ เช่น Least-Recently-Used (LRU) เป็นต้น
4. Request Forwarding: เป็นส่วนที่จะทำหน้าที่ส่งผ่านการร้องขอจากไคลเอนท์ไปยังเครื่องเว็บเซิร์ฟเวอร์อื่นหรือพร็อกซีเซิร์ฟเวอร์อื่นในบางกรณีที่มีการกำหนดไว้ล่วงหน้าแล้วที่ตัว Squid เช่น ให้ทำการส่งผ่านการร้องขอของไคลเอนท์ที่ต้องการร้องขอออบเจกต์จากเว็บเซิร์ฟเวอร์ที่อยู่ภายในโดเมนเดียวกันไปยังเว็บเซิร์ฟเวอร์นั้นโดยตรงเลย โดย Squid จะไม่ทำการร้องขอออบเจกต์ดังกล่าวให้เนื่องจากถือว่าการติดต่อภายในโดเมนเดียวกันนั้นสามารถทำได้อย่างรวดเร็วและมี Bandwidth ระหว่างเครื่องไคลเอนท์และเว็บเซิร์ฟเวอร์ต้นทางที่มากพอ (เช่นอาจจะอยู่บน Campus Network เดียวกัน)
5. Peer Selection: ทำหน้าที่ตัดสินใจในการเลือกหรือไม่เลือก พร็อกซีเซิร์ฟเวอร์ที่เหมาะสมที่จะทำการส่งผ่านการร้องขอออกไปในกรณีที่ไม่พบข้อมูลในแคชของตัวเอง
6. Access Control: ทำหน้าที่ในการรับหรือปฏิเสธการร้องขอจากไคลเอนท์ที่มีเข้ามาโดยสามารถพิจารณาจาก IP Address ของไคลเอนท์ ชื่อเครื่องเว็บเซิร์ฟเวอร์ที่เก็บออกเจกต์ที่ไคลเอนท์ ต้องการร้องขอ หรือ วิธีการในการร้องขอ (Request Method) โดย Squid สามารถรองรับการทำ Access Control List ได้หลายอันพร้อมกันโดยจะทำงานเรียงตามลำดับไปเรื่อย ๆ จนครบหมดทุก List
7. Network Communication: จะเป็นส่วนที่ทำหน้าที่ในการติดต่อสื่อสารผ่านเครือข่าย โดยสามารถติดต่อสื่อสารได้ทั้งในแบบ TCP และ UDP
8. File/Disk I/O: เป็นส่วนที่ทำหน้าที่ในการติดต่อกับดิสก์ในการอ่าน และเขียนข้อมูล โดยมันสามารถที่จะรวมความต้องการในการเขียนดิสก์จากหลาย ๆ การร้องขอเป็นการสั่งให้มีการเขียนดิสก์เพียงครั้งเดียวได้ เพื่อเป็นการเพิ่มประสิทธิภาพในการทำงาน โดยลดการเขียนข้อมูลขนาดเล็ก ๆ หลาย ๆ ครั้งลงเป็นการเขียนข้อมูลขนาดใหญ่ขึ้นด้วยจำนวนครั้งที่น้อยลง อันเป็นการลดการขัดจังหวะการทำงานของหน่วยประมวลผลกลางและเพิ่มประสิทธิภาพในการใช้งานดิสก์

9. Neighbor: ทำหน้าที่เกี่ยวกับการเก็บรายชื่อของ พร็อกซีเซิร์ฟเวอร์ที่ถูกกำหนดให้ทำงานร่วมกับตัวมัน ตลอดจนทำการรับ-ส่ง ICP Message ไปยังเครื่องพร็อกซีเซิร์ฟเวอร์ทั้งหมดที่อยู่ในรายชื่อนี้เพื่อใช้ประกอบการตัดสินใจเลือกพร็อกซีเซิร์ฟเวอร์ในการที่จะทำการร้องขอออบเจกต์ในกรณีที่ไม่พบออบเจกต์ที่ไคลเอนท์ร้องขอในแคชของตัวเอง
10. IP/FDQN: ทำหน้าที่เก็บและดูแลตารางความสัมพันธ์ระหว่าง ชื่อเครื่องเซิร์ฟเวอร์ที่ให้บริการกับ IP Address ของเครื่องนั้น เพื่อที่จะสามารถนำมาใช้ได้โดยไม่ต้องส่งการร้องขอไปยังเครื่องที่ให้บริการ DNS ซึ่งจะสามารถลดเวลาตรงนี้ลงไปได้ส่วนหนึ่ง
11. Cache Manager: ทำหน้าที่ในการจัดการกับตัว Squid ซึ่งเป็นพร็อกซีเซิร์ฟเวอร์อยู่โดยสามารถที่จะทำการจัดการและตรวจสอบข้อมูลจาก Squid หลาย ๆ เครื่องได้จากที่เดียวโดยผ่าน cachemgr.cgi ซึ่งมีหลักการทำงานคือตัวโปรแกรม cachemgr.cgi จะทำการส่ง URL ของออบเจกต์ที่ต้องการในรูปของ cache_object://hostname/operation ไปยังเครื่องพร็อกซีเซิร์ฟเวอร์ Squid ที่ต้องการจะตรวจสอบข้อมูล โดยจะทำได้เพียงการอ่านข้อมูลอย่างเดียวเท่านั้น ไม่สามารถที่จะทำการปรับตั้งค่าพารามิเตอร์ใด ๆ ได้ในขณะที่ Squid เป้าหมายนั้นทำงานอยู่
12. Network Measurement Database: จะทำหน้าที่ในการเก็บและดูแลข้อมูล round-trip time (RTT) ของตัว Squid กับพร็อกซีเซิร์ฟเวอร์อื่นข้างเคียงไปยังเครื่องเซิร์ฟเวอร์ต้นทาง เพื่อใช้ในการเปรียบเทียบเวลาว่าใครอยู่ใกล้เครื่องพร็อกซีเซิร์ฟเวอร์ต้นทางมากกว่ากัน เพื่อที่จะตัดสินใจเลือกว่าจะทำการร้องขอออบเจกต์จากเครื่องเซิร์ฟเวอร์ใด การวัดค่า RTT นี้สามารถทำได้โดยใช้โปรแกรม pinger ส่ง ICMP Packet ไปยังเครื่องปลายทางแล้ววัดเวลาที่ใช้ตั้งแต่เริ่มส่งจนได้รับการตอบกลับมาจากเครื่องปลายทาง
13. Redirector: ส่วนนี้มีความสามารถในการเขียนทับการร้องขอจากไคลเอนท์หลังจากที่ตรวจสอบพบว่ามีเงื่อนไขของการร้องขอตรงกับที่ระบุไว้ใน Access Control List โดยการร้องขอนี้จะถูกส่งออกไปยังแอพพลิเคชั่นตัวอื่นหรือแม้กระทั่งพร็อกซีเซิร์ฟเวอร์ตัวอื่นก็ได้ การทำ redirection นี้จะช่วยทำให้การใช้งานพร็อกซีเซิร์ฟเวอร์มีความหลากหลายและความยืดหยุ่นมากยิ่งขึ้น
14. Configuration File Parsing: เป็นส่วนที่ทำหน้าที่ตีความพารามิเตอร์ต่าง ๆ ที่ปรากฏอยู่ในไฟล์ squid.conf ซึ่งเป็นไฟล์ที่เราใช้กำหนดค่าเริ่มต้นและพารามิเตอร์ในการทำงานต่าง ๆ ของ Squid

เอกสารนี้เป็นเอกสารที่สงวน ... ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีก ... ใช้

15. Callback Data Database: เป็นส่วนที่ใช้จัดการฐานข้อมูลที่มีไว้เก็บและการเข้าถึงข้อมูลในหน่วยความจำหลักของเครื่องอันเนื่องมาจากการเรียกใช้ฟังก์ชั่น callback ซึ่ง Squid จะเรียกใช้บ่อยครั้งมากโดยจะเกิดขึ้นทุกครั้งที่มีการเขียนข้อมูลผ่านเครือข่าย (ในการเขียนข้อมูลลงดิสก์ก็เกิด callback ขึ้นเช่นกันหากแต่อาจจะไม่เกิดทุกครั้งเนื่องจากการทำงานในส่วนของ File/Disk I/O นั้นสามารถรวมการเขียนข้อมูลขนาดเล็กหลายครั้งเป็นการเขียนข้อมูลลงดิสก์จริงเพียงครั้งเดียวได้) การทำงานในส่วนนี้จะจัดเตรียมวิธีการในการจัดการกับข้อมูลที่เกี่ยวข้องกับการทำ callback ในหน่วยความจำ การยกเลิกการทำ callback และ การป้องกันความผิดพลาดอันเกิดจากความพยายามในการเข้าถึงข้อมูลจากหน่วยความจำหลัก
16. Debugging: เป็นส่วนที่ช่วยรายงานผลการทำงานเพื่อใช้ในการติดตามการทำงานของ Squid การตรวจหาสาเหตุของความผิดพลาดในการทำงานของ Squid โดยสามารถเลือกระดับของการแสดงผลการทำงานได้หลายระดับ
17. Error Generation: ทำหน้าที่ในการสร้างข้อความแสดงความผิดพลาด (Error Message) จากรูปแบบและพารามิเตอร์ที่กำหนด ซึ่งทำให้สามารถทำการปรับแต่งการแสดงความผิดได้ นอกจากนี้ยังสนับสนุนหลายภาษาอีกด้วย (Multilingual Support)
18. Event Queue: ทำหน้าที่เกี่ยวกับการจัดการเหตุการณ์ต่าง ๆ ที่เกิดขึ้นทุกช่วงระยะเวลา เช่นการทำ Cache Replacement การทำลบไฟล์ใน Swap Directory นอกจากนี้ยังสามารถใช้ได้กับการทำงานของฟังก์ชั่นที่ทำเพียงครั้งเดียว (one-time function) เช่น ICP query timeout
19. File descriptor Management: ทำหน้าที่ในการติดตามการใช้ File descriptor ตลอดจนจำนวนข้อมูลที่อ่านหรือเขียนไปยัง File descriptor แต่ละอัน
20. Hash Table Support ทำหน้าที่ในการสร้าง ตาราง Hash ในการสร้างความสัมพันธ์แบบ Hash ของออบเจกต์ที่เก็บในแคช
21. HTTP Anonymization เป็นส่วนที่ทำหน้าที่เกี่ยวกับการอนุญาตหรือไม่อนุญาตให้ Squid ประมวลผลการร้องขอโดยโพรโตคอล HTTP ที่มีส่วนหัว (Header) ตรงตามที่กำหนดไว้ ซึ่งโดยปรกติแล้วจะไม่มีการเรียกใช้การทำงานในส่วนนี้ นั่นคือจะยอมให้ Squid ทำการประมวลการร้องขอโดยโพรโตคอล HTTP โดยไม่มีข้อแม้ใด ๆ
22. Internet Cache Protocol: เป็นส่วนที่ทำหน้าที่เกี่ยวกับ ICP ทั้งหมดดังที่ได้กล่าวถึงมาแล้วข้างต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

23. Ident Lookups: เป็นส่วนที่ทำหน้าที่ในการร้องขอชื่อของผู้ใช้ในแต่ละการเชื่อมต่อที่มีเข้ามายัง Squid ซึ่งในบางที่ (Site) สามารถนำไปใช้ในการตรวจสอบสิทธิในการเข้าใช้งาน และการติดตามพฤติกรรมในการใช้งานเครื่องเซิร์ฟเวอร์ของที่นั่นได้อีกด้วย
24. Memory Management: ทำหน้าที่ในการกำหนดและจัดการเนื้อที่ในหน่วยความจำหลักสำหรับใช้ในการเก็บข้อมูลหรือออบเจกต์ที่ถูกเรียกใช้งานบ่อย ๆ เมื่อมีการเปิดการทำงานคำสั่ง memory_pool ในไฟล์ squid.conf พื้นที่ในหน่วยความจำที่ยังไม่มีการนำข้อมูลเข้าไปเก็บจะไม่ถือว่าเป็นพื้นที่ว่าง แต่จะถูกการทำงานในส่วนของ Memory Management กันที่จองเอาไว้สำหรับการใช้งานที่จะเกิดขึ้นในอนาคตต่อไป ซึ่งเป็นการใช้งานหน่วยความจำหลักอย่างไม่มีประสิทธิภาพนัก
25. Multicast support: การทำงานในส่วนนี้ในปัจจุบันจะใช้รองรับการส่ง ICP-query เท่านั้น โดยจะเชื่อม UDP socket เข้ากับ Multicast group และทำการกำหนดค่า TTL สำหรับการส่งแบบ multicast (ซึ่งจะมีค่าน้อยกว่าค่า TTL ของการส่งแบบ Unicast และ Broadcast)
26. Persistent Server Connections: จะทำหน้าที่จัดการกับการเชื่อมต่อแบบ Persistent ระหว่าง Squid ไปยังเครื่องเซิร์ฟเวอร์ต้นทางหรือ พร็อกซีเซิร์ฟเวอร์ใกล้เคียง โดยจะมีการเก็บค่าดัชนีของการเชื่อมต่อแต่ละการเชื่อมต่อไว้ใน Hash Table โดยการเก็บหมายเลข IP Address และ พอร์ต โดยที่ในแต่ละ socket address (IP Address + port) จะเก็บการเชื่อมต่อแบบ Persistent ได้มากที่สุด 10 การเชื่อมต่อ ทั้งนี้จะทำการรักษาการเชื่อมต่อดังกล่าวไว้เป็นเวลาไม่เกิน 15 วินาที หากไม่มีการใช้งานการเชื่อมต่อแบบ Persistent นั้นเกิน 15 วินาทีก็จะทำการปิดการเชื่อมต่อนั้นทันที
27. Refresh Rules: ทำหน้าที่ในการตัดสินใจว่าออบเจกต์ที่ถูกเก็บอยู่ในดิสก์แคชนั้นยังใช้ได้หรือหมดอายุแล้ว โดยขึ้นอยู่กับพารามิเตอร์ refresh_pattern ในไฟล์ squid.conf โดยในกรณีที่ออบเจกต์ที่ถูกร้องขอนั้นยังไม่หมดอายุ Squid ก็จะนำออบเจกต์นั้นส่งให้กับไคลเอนท์ที่ร้องขอเข้ามา และถือว่าการร้องขอครั้งนี้เป็นการร้องขอประเภทที่พบข้อมูล HIT แต่ถ้าออบเจกต์ที่ถูกร้องขอเข้ามาถูกตัดสินว่าหมดอายุแล้ว Squid ก็จะต้องทำการตรวจสอบความถูกต้องของออบเจกต์นั้นกับเซิร์ฟเวอร์ต้นทางด้วยการร้องขอแบบ If-Modify-Since

เอกสารนี้เป็นเอกสารที่สงวน ไว้ ... ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้าม ... ไปใช้

28. SNMP Support: เป็นส่วนที่ใช้สนับสนุนการทำงานร่วมกับโพรโตคอลSNMP โดยปัจจุบัน Squid สามารถให้ข้อมูลเกือบทั้งหมดที่สามารถหาได้จากโปรแกรม cachemgr ผ่านทาง SNMP ได้
29. URN Support: เป็นส่วนที่ทำหน้าที่ในการสนับสนุนและรองรับการระบุออบเจกต์และประเภทของการบริการ แบบ Uniform Resource Name (URN) ซึ่งเป็นรูปแบบการระบุออบเจกต์ที่คล้ายคลึงกับ URL

2.1.2 ขั้นตอนการทำงานของ Squid พร็อกซีเซิร์ฟเวอร์

การทำงานของพร็อกซีเซิร์ฟเวอร์ในที่นี้จะหมายถึงการทำงานที่เกิดจากการสั่งงานของ Squid ซึ่งเป็นแอพพลิเคชั่นที่ทำหน้าที่เป็นพร็อกซีอยู่บนตัวเครื่องพร็อกซีเซิร์ฟเวอร์ โดยแบ่งได้เป็นสองช่วง คือ ช่วงเริ่มต้นระบบ (System Initialization Phase) และช่วงให้บริการ (Service Phase)

ช่วงเริ่มต้นระบบ (System Initialize Phase)

ถ้าเป็นการเริ่มต้นการทำงานครั้งแรกตัวพร็อกซีเซิร์ฟเวอร์จะทำการเตรียมพื้นที่เก็บข้อมูลบน ดิสก์ โดยการจัดสร้างไดเรกทอรีและไดเรกทอรีย่อยตามที่ถูกตั้งค่าเอาไว้แล้วจึงเข้าสู่ช่วงการให้บริการ

ถ้าเป็นการเริ่มต้นการทำงานครั้งต่อ ๆ ไปพร็อกซีเซิร์ฟเวอร์จะทำการตรวจสอบความถูกต้องของฐานข้อมูลที่เกี่ยวกับออบเจกต์ที่เก็บให้ในดิสก์แคชตลอดจนตรวจสอบและลบออบเจกต์ที่มีอายุ (ที่เก็บไว้ใน ดิสก์แคช) เกินที่ตั้งค่าเอาไว้ จากนั้นจึงเข้าสู่ช่วงการให้บริการ

ช่วงให้บริการ (Service Phase)

การทำงานจะเริ่มจาก

1. รับการร้องขอ: เมื่อเครื่องพร็อกซีเซิร์ฟเวอร์ได้รับ TCP Connection จากเครื่องไคลเอนท์ผ่านทางพอร์ตที่กำหนด (ส่วนมากจะเป็น พอร์ต 8080 ยกเว้น Squid ที่ใช้ พอร์ต 3128) โดยไคลเอนท์จะส่งการร้องขอเข้ามายังพร็อกซีเซิร์ฟเวอร์ซึ่งประกอบไปด้วย URL ของออบเจกต์และวิธีที่ต้องการกระทำกับออบเจกต์นั้น ๆ ตามมาตรฐานของ HTTP/FTP เช่น GET, POST, REFRESH เป็นต้น

2. ค้นหาออบเจกต์ในเมโมรีแคช: เมื่อได้รับ การร้องขอ เข้ามาพร็อกซีเซิร์ฟเวอร์จะทำการตรวจสอบออบเจกต์ที่ได้รับการร้องขอนั้นมีอยู่ในเมโมรีแคชหรือไม่ ถ้าพบออบเจกต์ ดังกล่าว (MEM_HIT) ก็จะทำการส่งออบเจกต์นั้นให้กับไคลเอนท์แต่ถ้าไม่พบก็จะทำการค้นหาออบเจกต์เหล่านั้นใน ดิสก์แคช ต่อไป

3. ค้นหาออบเจกต์ในดิสก์แคช: ในกรณีที่พร็อกซีเซิร์ฟเวอร์ไม่พบออบเจกต์ใน เมโมรีแคชก็จะทำการตรวจสอบออบเจกต์ที่ได้รับการร้องขอนั้นว่ามีเก็บอยู่ในดิสก์แคชของตัวเองหรือไม่

3.1 Request HIT: ถ้าออบเจกต์ดังกล่าวมีอยู่ในดิสก์แคชและมีอายุไม่เกินที่ตั้งค่าไว้ก็จะทำการอ่านออบเจกต์ขึ้นมาไว้ในเมโมรีแคชและทำการส่งออบเจกต์ดังกล่าวไปให้กับไคลเอนท์ที่ได้ร้องขอมา

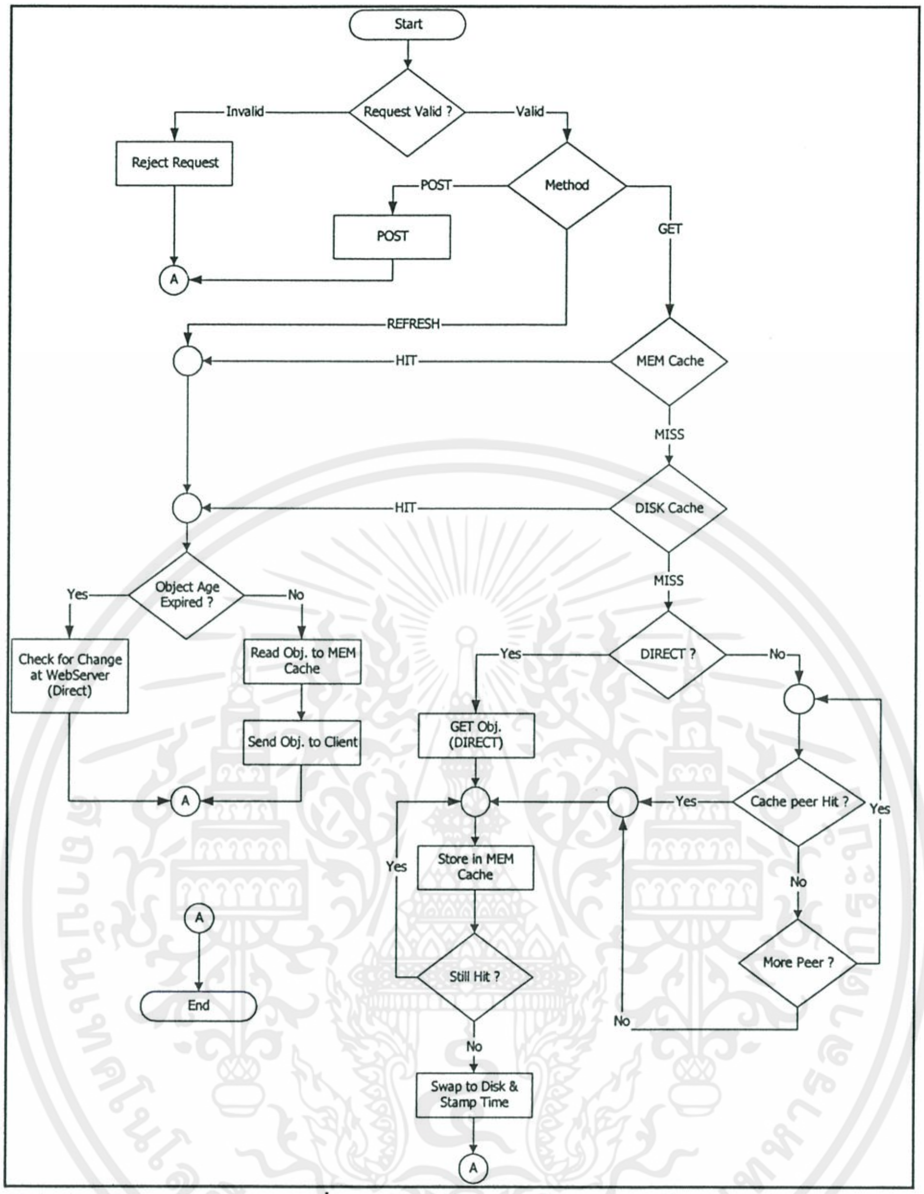
3.2 Request MISS: ถ้าไม่พบออบเจกต์ในดิสก์แคช ตัวพร็อกซีเซิร์ฟเวอร์เองจะมีทางเลือกสองทางคือ

3.2.1 DIRECT: ตัวพร็อกซีเซิร์ฟเวอร์จะทำการติดต่อไปร้องขอออบเจกต์นั้นจากเว็บเซิร์ฟเวอร์ตาม URL ที่ระบุไว้ในการร้องขอโดยตรง โดยเมื่อได้ออบเจกต์ดังกล่าวมาแล้วก็จะทำการจัดเก็บไว้ในเมโมรีแคช และทำการส่งให้กับไคลเอนท์ที่ร้องขอมา

3.2.2 CACHE PEER: พร็อกซีเซิร์ฟเวอร์จะสร้างการติดต่อไปยังพร็อกซีเซิร์ฟเวอร์เครื่องอื่นในระบบ โดยวิธีการที่จะใช้ขึ้นกับความสัมพันธ์ระหว่างพร็อกซีเซิร์ฟเวอร์คู่นั้น เช่นถ้าทั้งสองเป็น Sibling ซึ่งกันและกัน (อยู่ระดับเดียวกัน) ในกรณีของ Squid จะใช้โพรโตคอล ICP ผ่านทาง UDP พอร์ต 3130 แต่ถ้าพร็อกซีเซิร์ฟเวอร์นั้นกับมีความสัมพันธ์แบบ Parent-Child กับพร็อกซีเซิร์ฟเวอร์ที่จะร้องขอบริการพร็อกซีเซิร์ฟเวอร์ก็จะทำหน้าที่เหมือนกับเป็นไคลเอนท์ของพร็อกซีเซิร์ฟเวอร์ ที่มันไปขอใช้บริการ ซึ่งผลของการร้องขอก็จะมีทั้งพบออบเจกต์ (PARENT_HIT, SIBLING_HIT) และไม่พบ (PARENT_MISS, UDP_MISS) โดยถ้าพบ ออบเจกต์ ก็จะนำมาเก็บไว้ใน เมโมรีแคช แล้วจึงจัดส่งให้ ไคลเอนท์ต่อไป แต่ถ้าไม่พบออบเจกต์ดังกล่าวก็จะทำติดต่อไปยังพร็อกซีเซิร์ฟเวอร์เครื่องต่อไปตามที่ตั้งค่าไว้จนครบ ในกรณีที่ไม่พบออบเจกต์ที่ต้องการจากเครื่องพร็อกซีเซิร์ฟเวอร์ทุกเครื่องที่ตั้งค่าไว้ก็จะทำการติดต่อไปยังเว็บเซิร์ฟเวอร์ตาม URL ของออบเจกต์นั้น

4. บันทึกผลการทำงาน: เมื่อพร็อกซีเซิร์ฟเวอร์ได้รับการร้องขอเข้ามาและทำการประมวลและทำงานตามการร้องขอนั้นจนเสร็จสิ้นการร้องขอนั้นแล้วก็จะทำการบันทึกผลการทำงานของล็อกไฟล์ ซึ่งจะได้กล่าวถึงรายละเอียดของล็อกไฟล์อีกครั้งหนึ่ง

เอกสารนี้เป็นเอ ... ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 โฟลว์ชาร์ตสรุปการทำงานเมื่อ Squid ได้รับการร้องขอจากไคลเอนท์

จากขั้นตอนการทำงานของ Squid ข้างต้นสามารถนำมาแยกผลของการร้องขอเพื่ออธิบายผลของการร้องขอแต่ละประเภทเพื่อแสดงถึงการไหลของพารามิเตอร์ที่เกี่ยวข้องรวมถึงองค์ประกอบในการทำงานต่าง ๆ ของ Squid โดยอาศัยแผนภาพที่อธิบายขั้นตอนการทำงานในกรณีต่าง ๆ ดังต่อไปนี้

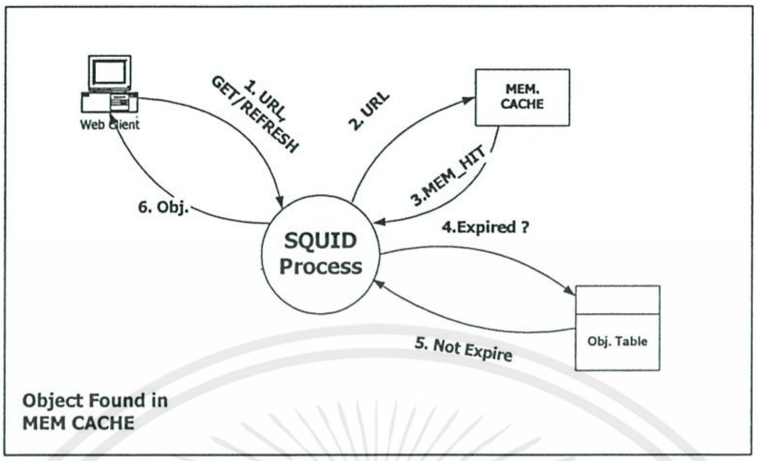
1. กรณีที่พบออบเจกต์ที่ร้องขอในเมโมรีแคช

2. กรณีที่ไม่พบออบเจกต์ในเมโมรีแคชแต่พบออบเจกต์ที่ร้องขอในดิสก์แคช

3. กรณีที่ไม่พบออบเจกต์ในเมโมรีแคชและดิสก์แคช

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. กรณีที่พบออบเจกต์ที่ร้องขอในเมโมรีแคช



รูปที่ 2.2 ผลของการร้องขอที่พบออบเจกต์ในเมโมรีแคช

ในรูปที่ 2.2 การทำงานของ Squid ในกรณีที่พบข้อมูลในเมโมรีแคชจะมีลำดับในการทำงานดังต่อไปนี้

1. มีการร้องขอจากไคลเอนท์เข้ามา โดยไคลเอนท์จะส่ง URL ของออบเจกต์และวิธีการที่ต้องการให้พร็อกซีเซิร์ฟเวอร์กระทำต่อออบเจกต์นั้น เช่น สั่งให้นำมา (GET) สั่งให้ตรวจสอบการเปลี่ยนแปลง (REFRESH) ส่วนคำสั่งให้ส่งออบเจกต์ออกไป (POST) นั้นจำ Squid จะทำการส่งออกไปทันทีโดยไม่ได้จัดเก็บออบเจกต์ดังกล่าวไว้ในดิสก์แคช

2. เมื่อได้ URL มาแล้ว Squid จะทำการตรวจสอบว่ามีออบเจกต์ที่ระบุนั้นในเมโมรีแคชหรือไม่

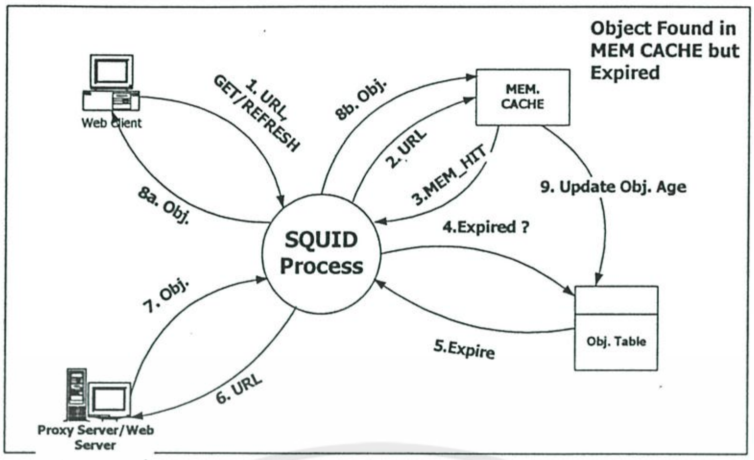
3. ผลของการค้นหาปรากฏว่าพบออบเจกต์ในเมโมรีแคช

4. ทำการตรวจสอบว่าออบเจกต์นั้นมีอายุในการเก็บไว้ในแคชเกินเวลาที่กำหนดไว้หรือไม่

5. ผลของการตรวจสอบพบว่าออบเจกต์นั้นมีอายุในแคชยังไม่เกินเวลาที่กำหนด

6. Squid ทำการส่งออบเจกต์นั้นไปให้ไคลเอนท์ตามคำร้องขอ

ในกรณีที่ผลของการตรวจสอบว่าออบเจกต์ที่พบในเมโมรีแคชนั้นมีอายุเกินกว่าที่กำหนดไว้ Squid ก็จะทำการร้องขอออบเจกต์นั้นไปยังเซิร์ฟเวอร์ต้นทางตาม URL ของออบเจกต์นั้นโดยตรง หรือถ้าหาก Squid ถูกกำหนดค่าให้มีการทำงานร่วมกับเครื่องพร็อกซีเซิร์ฟเวอร์เครื่องอื่นก็จะทำการร้องขอไปยังพร็อกซีเซิร์ฟเวอร์ดังกล่าวตามที่ได้กำหนดค่าเอาไว้ ซึ่งจะมีขั้นตอนเพิ่มตั้งแต่ขั้นตอนที่ 5 ดังแสดงในรูปที่ 2.3



รูปที่ 2.3 ผลของการร้องขอที่พบออบเจกต์ในเมโมรีแคชแต่หมดอายุ

5. ผลการตรวจสอบปรากฏว่าออบเจกต์มีอายุที่เก็บในแคชเกินที่กำหนดไว้

6. Squid จะทำการร้องขอออบเจกต์ตาม URL ที่ระบุ หรือขอตรวจสอบว่าออบเจกต์มีการเปลี่ยนแลงหรือไม่ไปยังเซิร์ฟเวอร์ต้นทางหรือ พร็อกซีเซิร์ฟเวอร์ที่ได้มีการกำหนดค่าไว้

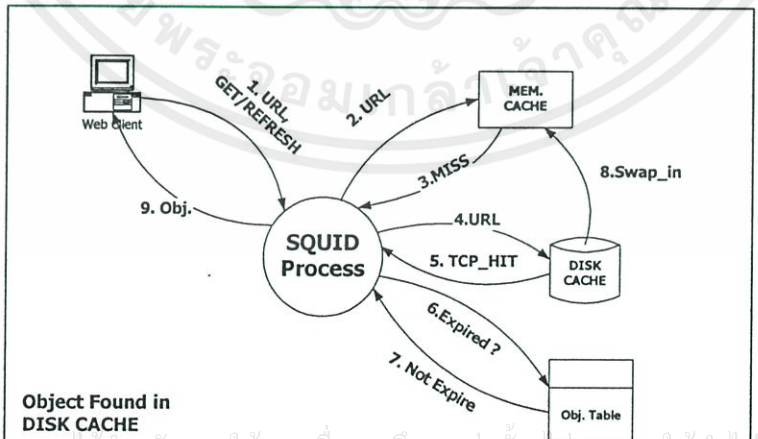
7. ได้รับออบเจกต์จากเซิร์ฟเวอร์ที่ร้องขอไปในขั้นตอนที่ 6

8a. Squid จะทำการส่งออบเจกต์ไปให้กับไคลเอนท์ที่ร้องขอออบเจกต์นั้น

8b. Squid ทำการทำสำเนาออบเจกต์นั้นไปเก็บไว้ในเมโมรีแคช

9. ทำการปรับปรุง (Update) ค่าอายุของออบเจกต์ที่เก็บในแคช

2. กรณีที่ไม่พบออบเจกต์ในเมโมรีแคชแต่พบออบเจกต์ที่ร้องขอในดิสก์แคช



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 2.4 ผลของการร้องขอที่พบออบเจกต์ในดิสก์แคช

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ในรูปที่ 2.4 การทำงานของ Squid ในกรณีที่พบข้อมูลในดิสก์แคชจะเริ่มจากรับการร้องขอจากไคลเอนท์เข้ามาและทำการตรวจสอบว่ามีออบเจกต์ดังกล่าวในเมโมรีแคชหรือไม่ (ขั้นตอนที่ 1 และ 2) แต่กรณีนี้ไม่พบออบเจกต์ที่ต้องการ จึงมีขั้นตอนเพิ่มเติมดังต่อไปนี้

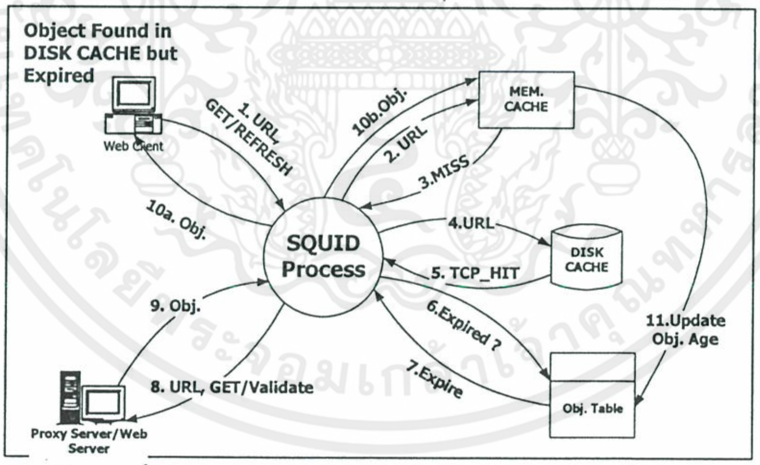
3.ผลการตรวจสอบปรากฏว่าไม่พบออบเจกต์ในเมโมรีแคช

4.Squid จะทำการตรวจสอบว่ามีออบเจกต์ตาม URL ในดิสก์แคชหรือไม่

5.ผลการตรวจสอบปรากฏว่าพบออบเจกต์ในดิสก์แคช

จากนั้น Squid ก็จะทำการตรวจสอบอายุของออบเจกต์ที่เก็บไว้ในแคชว่าเกินค่ากำหนดที่ตั้งไว้หรือไม่ ถ้าผลของการตรวจสอบปรากฏว่าออบเจกต์ยังมีอายุไม่เกินค่าที่กำหนด ก็จะทำการสำเนาออบเจกต์นั้นไปเก็บไว้ในเมโมรีแคช (ขั้นตอนที่ 8) และส่งออบเจกต์นั้นให้กับไคลเอนท์ที่ร้องขอเข้ามา (ขั้นตอนที่ 9)

ในกรณีที่พบออบเจกต์ในดิสก์แคชแต่ปรากฏว่าออบเจกต์นั้นมีอายุที่เก็บไว้ในแคชนานเกินกำหนดที่ตั้งไว้ ก็จะต้องเพิ่มขั้นตอนในการไปร้องขอออบเจกต์ (GET) หรือขอตรวจสอบว่าออบเจกต์ดังกล่าวมีการเปลี่ยนแปลงหรือไม่ (REFRESH) ดังกล่าว ดังแสดงไว้ในขั้นตอนที่ 8 และ 9 ในรูปที่ 2.5 ซึ่งเมื่อ Squid ได้รับการตอบรับกลับมาแล้วก็จะทำการส่งออบเจกต์นั้นให้กับไคลเอนท์ที่ร้องขอเข้ามาและทำการปรับปรุงค่าอายุของออบเจกต์นั้นที่เก็บไว้ในแคชใหม่

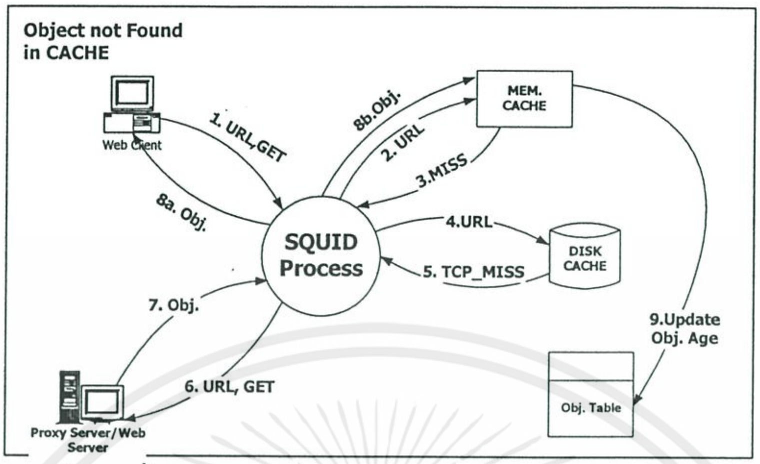


รูปที่ 2.5 ผลของการร้องขอที่พบออบเจกต์ในดิสก์แคชแต่หมดอายุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

40006

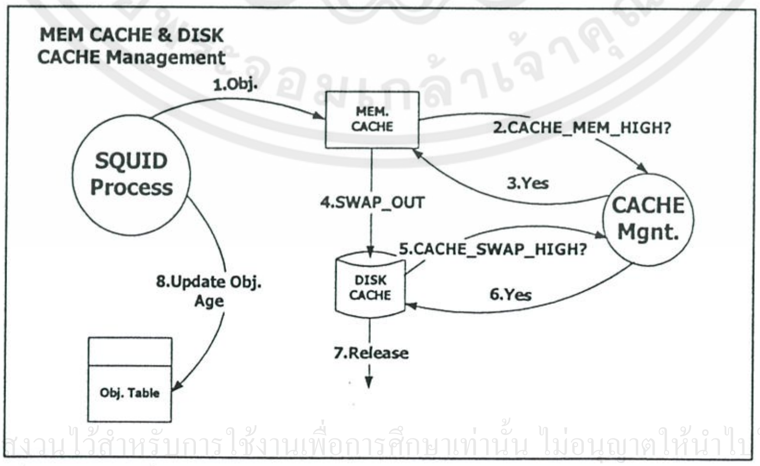
3. กรณีที่ไม่พบออบเจกต์ในเมโมรีแคชและดิสก์แคช



รูปที่ 2.6 ผลของการร้องขอที่ไม่พบออบเจกต์ในแคชของ Squid

ในกรณีที่ออบเจกต์ที่ถูกร้องขอเข้ามานั้นไม่พบอยู่ทั้งในเมโมรีแคชและดิสก์แคชนั้น Squid ก็จะทำการร้องขอออบเจกต์นั้นไปยังเซิร์ฟเวอร์ต้นทางหรือพร็อกซีเซิร์ฟเวอร์ที่ได้มีการตั้งค่าเอาไว้ ซึ่งการทำงานก็จะคลายกับกรณีที่ Squid เจอออบเจกต์ที่ถูกร้องขอในแคชของตัวเองแต่ปรากฏว่าออบเจกต์นั้นหมดอายุ เพียงแต่ในขั้นตอนที่ 6 จากรูปที่ 2.6 จะเป็นการร้องขอแบบขอออบเจกต์ (GET) ไม่ใช่การตรวจสอบว่าออบเจกต์นั้นมีการเปลี่ยนแปลงหรือไม่ (REFRESH)

ในการทำงานของ Squid จะมีการเก็บออบเจกต์ใหม่ที่ถูกร้องขอเข้ามาไว้ในแคชโดยจะมีลำดับขั้นตอนการทำงานและส่วนที่เกี่ยวข้องดังแสดงไว้ในรูปที่ 2.7



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 2.7 การจัดเก็บออกเจกต์ในแคชของ Squid

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เมื่อ Squid ได้รับออบเจกต์ที่ถูกร้องขออันใหม่เข้ามาก็นำมาจัดเก็บไว้ในส่วนของเมโมรีแคชเป็นอันดับแรก

2. ถึง 4.เมื่อพื้นที่ที่ใช้ในการจัดเก็บออบเจกต์ของเมโมรีแคชมีค่ามากถึงค่าระดับที่ตั้งไว้ก็จะมีการย้ายออบเจกต์จำนวนหนึ่งจากเมโมรีแคชลงไปเก็บไว้ในดิสก์แคช

5. ถึง 7.ในทำนองเดียวกันเมื่อขนาดของพื้นที่ที่ใช้จัดเก็บออบเจกต์ของดิสก์แคชมีค่าถึงระดับที่ตั้งไว้ก็จะมีการลบออบเจกต์จำนวนหนึ่งทิ้งไปเพื่อให้มีเนื้อที่เหลือพอสำหรับจัดเก็บออบเจกต์ที่จะเข้ามาใหม่

8. เมื่อมีการนำออบเจกต์ใหม่เข้ามาเก็บก็จะมีการเก็บค่าอายุที่ออบเจกต์นั้นถูกเก็บไว้ในแคชด้วย

ในกรณีที่แคชเก็บข้อมูลจนถึงระดับที่ตั้งไว้ก็จะต้องมีกระบวนการเพื่อใช้ในการตัดสินใจที่จะทำการแทนที่ข้อมูลเก่าซึ่งมีอัลกอริธึมในการแทนที่ข้อมูลอยู่หลายวิธีด้วยกัน [18] ตัวอย่างเช่น

- Least-Recently Used (LRU) จะทำการเลือกแทนที่ข้อมูลที่ไม่ได้ถูกเรียกใช้จากไคลเอนท์นานที่สุด
- Least-Frequently-Used (LFU) จะทำการเลือกแทนที่ข้อมูลที่มีความถี่ในการเรียกใช้จากไคลเอนท์น้อยที่สุด
- Size จะทำการเลือกแทนที่ข้อมูลที่มีขนาดใหญ่ที่สุดเป็นอันดับแรก
- LRU-Threshold จะใช้หลักในการเลือกแทนที่ออบเจกต์ในแคชเหมือนกับวิธีแบบ LRU แต่มีเกณฑ์ของขนาดสูงสุดของออบเจกต์ที่จะเก็บไว้ในแคชกำกับไว้ด้วย โดยถ้าออบเจกต์ที่จะทำการเก็บมีขนาดใหญ่กว่าเกณฑ์ที่กำหนดก็จะไม่เก็บออบเจกต์นั้นลงในแคช
- Log (Size) + LRU จะใช้หลักในการเลือกแทนที่ออบเจกต์ในแคชเหมือนกับวิธีแบบ LRU แต่จะทำการเปรียบเทียบเฉพาะกับออบเจกต์ที่มีขนาดใกล้เคียงกับข้อมูลที่จะนำมาแทนที่เท่านั้น
- Hyper-G จะใช้หลักในการเลือกแทนที่ออบเจกต์ในแคชเหมือนกับวิธีแบบ LFU แต่จะนำ เวลาในการเรียกใช้ออบเจกต์ครั้งล่าสุดและขนาดมาพิจารณาประกอบ
- Pitkow/Rocker จะใช้หลักในการเลือกแทนที่ออบเจกต์ในแคชเหมือนกับวิธีแบบ LRU ยกเว้นในกรณีที่ออบเจกต์ทั้งหมดถูกร้องขอภายในวันเดียวกันทั้งหมด (ไม่มีออบเจกต์ใดไม่ถูกร้องขอนานเกินกว่า 1 วัน) จะเปลี่ยนมาพิจารณาแทนที่ออบเจกต์ที่มีขนาดใหญ่ที่สุด

เอกสารนี้เป็นเอกสารที่ส ... ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีก ... หา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Lowest-Latency-First วิธีนี้แคชจะพยายามลดค่าเวลาแฝง (Latency time) ในการเรียกข้อมูลจากเว็บเซิร์ฟเวอร์ต้นทาง โดยจะทำการแทนที่ออบเจกต์ที่มีเวลาแฝงในการเรียกข้อมูลน้อยที่สุดก่อนเป็นอันดับแรก
- Least Frequently Used with Dynamic Aging (LFUDA) [12], [15] วิธีนี้จะทำงานคล้ายกับ LFU แต่มันจะบวกอายุของออบเจกต์ที่ถูกเรียกใช้เข้ากับกับจำนวนครั้งที่ออบเจกต์นี้ถูกเรียกใช้ด้วย และจะเลือกออบเจกต์ที่มีค่าจำนวนการถูกเรียกใช้ซึ่งบวกกับค่าอายุของมันแล้วมีค่าน้อยที่สุดออกไปจากดิสก์แคช
- Greedy-Dual Size Frequency (GDSF) [12], [15] วิธีนี้จะนำขนาดของ ออบเจกต์มาคำนวณร่วมกับจำนวนการเรียกใช้งานและอายุของออบเจกต์ในแคช โดยคำนวณจาก

$$K = \frac{C}{S} + L \tag{2.1}$$

โดยที่

K คือค่าที่ Squid จะนำไปใช้ในการตัดสินใจเลือกลบออบเจกต์นั้นออกไปโดยจะลบออบเจกต์ที่มีค่านี้ต่ำที่สุดก่อน

C คือจำนวนครั้งที่ออบเจกต์นี้ถูกเรียกใช้งาน

S คือขนาดของออบเจกต์

L คือเวลาที่ออบเจกต์ถูกเก็บอยู่ในแคช

2.2 รูปแบบความสัมพันธ์ระหว่างพร็อกซีเซิร์ฟเวอร์

เราสามารถแบ่งประเภทของพร็อกซีเซิร์ฟเวอร์ตามลักษณะการจัดวางรูปแบบของแคชตามโทโปโลยี (Topology) ได้เป็นสองประเภท คือ Simple Web-cache และ Cooperate Web-Cache [1]

1. Simple Web-cache ซึ่งหมายถึงการเชื่อมต่อพร็อกซีเซิร์ฟเวอร์ที่มีการทำงานต่อกันเป็นลำดับชั้น (Hierarchy) เมื่อไม่พบออบเจกต์ที่แคชของตัวเองก็จะทำการร้องขอออบเจกต์ดังกล่าวกับพร็อกซีเซิร์ฟเวอร์เครื่องอื่นที่อยู่ในระดับสูงขึ้นไป

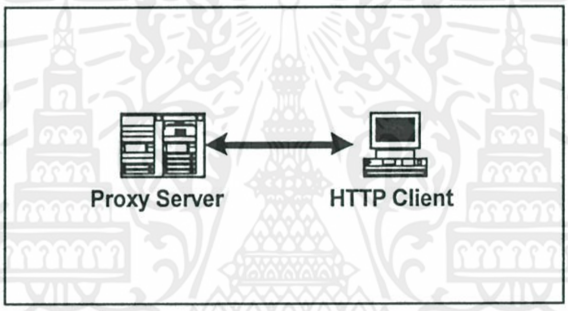
2. Cooperate Web-Cache สามารถแบ่งกระจายความรับผิดชอบในการให้ตอบสนองต่อออบเจกต์ที่ถูกร้องขอจากเซิร์ฟเวอร์ต้นทางที่อยู่ภายใต้กลุ่มของโดเมนเนมที่กำหนด เพื่อเป็นการลดความซ้ำซ้อนในการทำสำเนาออบเจกต์ในแต่ละดิสก์แคชของพร็อกซีเซิร์ฟเวอร์

เอกสาร ... ไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเราสามารถอธิบายความสัมพันธ์ระหว่างเครื่องพร็อกซีเซิร์ฟเวอร์แต่ละเครื่องได้ดังนี้

1. Single Proxy

เป็นระบบที่มีพร็อกซีเซิร์ฟเวอร์เพียงตัวเดียวโดยไคลเอนท์ทั้งหมดที่จะติดต่อขอใช้บริการจากเครื่องเซิร์ฟเวอร์นี้เพียงเครื่องเดียว เป็นระบบที่ง่ายต่อการติดตั้งและปรับแต่งค่าทั้งทางฝั่งเซิร์ฟเวอร์ และไคลเอนท์ ตลอดจนการควบคุมสิทธิในการใช้งานของผู้ใช้แต่การที่มีเครื่องให้บริการเพียงเครื่องเดียวอาจจะเสี่ยงแต่ความล้มเหลวของระบบ (Single point of failure) และปัญหาคอขวดที่เครื่องพร็อกซีเซิร์ฟเวอร์มีความสามารถในการให้บริการไม่เพียงพอต่อปริมาณความต้องการของไคลเอนท์ นอกจากนี้การออกแบบระบบให้มีเครื่องพร็อกซีเซิร์ฟเวอร์เพียงเครื่องเดียวนั้นทำให้ต้องใช้เครื่องที่มีประสิทธ์ภาพในการทำงานและทรัพยากรที่สูงซึ่งหมายความว่าจะต้องใช้งปประมาณที่สูงมากสำหรับเครื่องเซิร์ฟเวอร์นี้

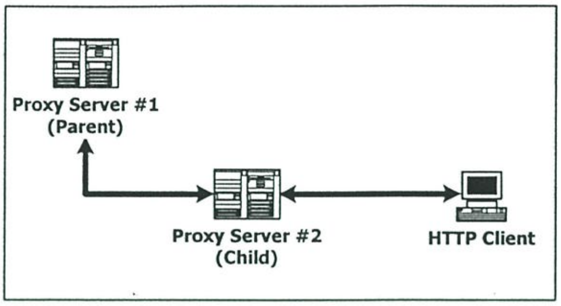


รูปที่ 2.8 ระบบพร็อกซีเซิร์ฟเวอร์แบบ Single Proxy

2. Parent-Child Relation

เป็นระบบที่ประกอบไปด้วย พร็อกซีเซิร์ฟเวอร์ ตั้งแต่สองเครื่องขึ้นไป โดยจะมีเครื่องพร็อกซีเซิร์ฟเวอร์ หลักที่มีขนาดใหญ่และประสิทธิภาพสูงอยู่ที่ระดับบนสุด (Parent) ทำหน้าที่คอยรับการร้องขอบริการจากเครื่องพร็อกซีเซิร์ฟเวอร์ที่อยู่ระดับต่ำกว่า (Child) ซึ่งโดยมากจะเป็นเครื่องพร็อกซีเซิร์ฟเวอร์ ที่มีขนาดและประสิทธิภาพต่ำกว่าอยู่ในระดับถัดมาเพื่อทำหน้าที่รับการร้องขอบริการจากเครื่อง ไคลเอนท์ ทั้งนี้จำนวนของเครื่อง พร็อกซีเซิร์ฟเวอร์ ที่ทำหน้าที่เป็น Child นั้นสามารถมีได้หลายเครื่องเพื่อเป็นการกระจายภาระงานไปยังพร็อกซีเซิร์ฟเวอร์ หลายตัวและเข้าถึงกลุ่มของ ไคลเอนท์ได้อย่างหลากหลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

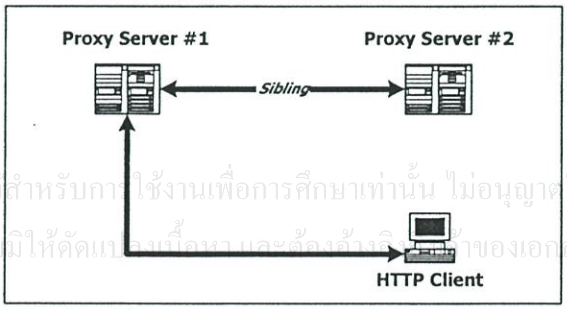


รูปที่ 2.9 ระบบพร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์แบบ Parent-Child

ระบบพร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์แบบนี้เป็นตัวอย่างของระบบแบบ Simple Web-Cache นั้นคือเครื่องพร็อกซีเซิร์ฟเวอร์หมายเลข 2 ในรูปที่ 2.9 จะต้องทำการร้องขอข้อมูลผ่านเครื่องพร็อกซีเซิร์ฟเวอร์หมายเลข 1 ทุกครั้งที่ไม่พบข้อมูลในแคชที่ตัวเอง และเมื่อเครื่องหมายเลข 1 นำออบเจกต์ที่ถูกร้องขอส่งมาให้เครื่องหมายเลข 2 นั้นตัวเครื่องหมายเลข 1 เองก็จะทำการสำเนาออบเจกต์ดังกล่าวไว้ด้วย ตรงนี้เองที่ทำให้เกิดความซ้ำซ้อนในการเก็บข้อมูล ยิ่งถ้าในกรณีที่ในระบบมีพร็อกซีเซิร์ฟเวอร์หลายเครื่องและมีความสัมพันธ์แบบ Parent-Child ลดหลั่นลงมาเป็นลำดับชั้น (Hierarchy) ก็จะทำให้เกิดความซ้ำซ้อนในการเก็บสำเนาออบเจกต์มาขึ้นตามไปด้วย [11]

3. Sibling Relation

เป็นระบบที่มีพร็อกซีเซิร์ฟเวอร์มากกว่าหนึ่งตัวประกอบกันขึ้นเพื่อรองรับการร้องขอใช้บริการของไคลเอนท์โดยเครื่องพร็อกซีเซิร์ฟเวอร์แต่ละตัวจะทำการร้องถามไปยังพร็อกซีเซิร์ฟเวอร์ที่เป็น Sibling ของมันว่ามีข้อมูลที่ต้องการเก็บไว้หรือไม่พร้อมทั้งตรวจสอบเวลาในการไปร้องขอออบเจกต์ที่อยู่ที่เครื่องเซิร์ฟเวอร์ต้นทางจากตัวเองเปรียบเทียบกับจากพร็อกซีเซิร์ฟเวอร์ที่เป็น Sibling ของมันเพื่อใช้ในการตัดสินใจ เครื่องพร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์แบบ Sibling นี้จะถือว่ามีความสัมพันธ์อยู่ในระดับ (Level) เดียวกันและสามารถปรับตั้งได้ว่าในกรณีที่พร็อกซีเซอร์เวอร์ตัดสินใจร้องขอข้อมูลจาก Sibling ของมันแล้วจะให้ทำสำเนาหรือไม่ทำสำเนาออบเจกต์นั้นเก็บลงในดิสก์แคชของตัวเองก็ได้



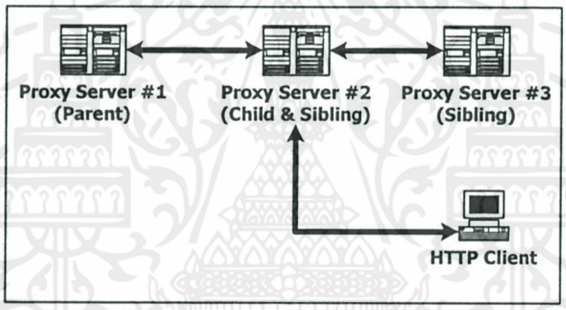
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงที่มาของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.10 ระบบพร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์แบบ Sibling

ระบบพร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์แบบนี้เป็นตัวอย่างของ Cooperate Web-Cache โดยที่เราสามารถปรับตั้งให้พร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์กันแบบ Sibling นี้ไม่ต้องทำการสำเนาออบเจกต์ที่ร้องขอมาได้จาก Sibling ของตัวเอง (ไม่ใช่ร้องขอมาได้จากเซิร์ฟเวอร์ต้นทาง) ซึ่งจะไม่ทำให้เกิดความซ้ำซ้อนของการจำเก็บออบเจกต์ขึ้น

4. Compound Relation

เป็นระบบที่ประกอบไปด้วย พร็อกซีเซิร์ฟเวอร์ จำนวนมาก ที่มีความสัมพันธ์กันทั้งแบบ Parent-Child และ Sibling ซึ่งสามารถจะรองรับปริมาณการร้องขอใช้บริการของ HTTP Client ได้สูงและมีความอ่อนตัวในการปรับแต่ง Proxy แต่ละตัวเพื่อทำสมดุลภาระงาน (Load-balancing) ซึ่งสามารถเพิ่มจำนวนของเครื่องพร็อกซีเซิร์ฟเวอร์ได้เมื่อมีความต้องการเพิ่มมากขึ้น ตลอดจนสามารถใช้แนวคิดของ Cooperate Web-Cache เพื่อละความซ้ำซ้อนของการเก็บข้อมูลได้อีกด้วย



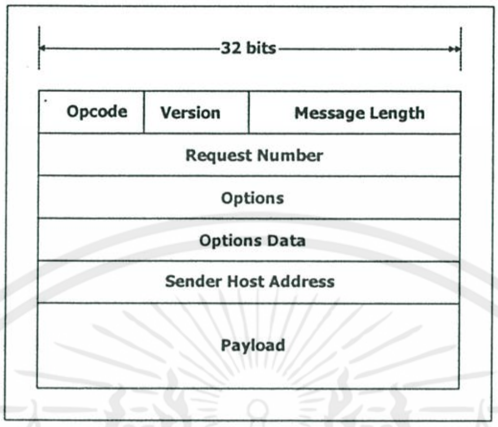
รูปที่ 2.11 ตัวอย่างระบบพร็อกซีเซิร์ฟเวอร์ที่มีความสัมพันธ์แบบผสม

ในระบบที่มีพร็อกซีทำงานร่วมกันตั้งแต่ 2 เครื่องขึ้นไปจะมีการใช้โพรโตคอล ICP ที่ใช้ในการสื่อสารระหว่างเว็บพร็อกซี/แคช เพื่อแลกเปลี่ยนข้อมูลเกี่ยวกับ URLs ที่มีอยู่ในแคชใกล้เคียง โดยจะเป็นการแลกเปลี่ยน ICP message (ICP-Query & ICP-Reply) เพื่อรวบรวมข้อมูลที่ใช้ในการร้องขอออบเจกต์จากแคชที่อยู่ใกล้เคียง โดยแคชจะส่ง ICP-Query ไปยังแคชที่อยู่ใกล้เคียง เมื่อได้รับแล้วจึงตอบกลับมาด้วย ICP-Reply เพื่อตอบว่า พบ (Hit) หรือไม่พบ (MISS) ออบเจกต์ที่ร้องขอนั้น โดยการส่ง ICP-message นี้ จะส่งโดยเลือกใช้บริการจาก TCP หรือ UDP ก็ได้ แต่การใช้บริการจาก UDP จะมีความเหมาะสมมากกว่าเพราะไม่มีการสร้าง Connection จึงทำงานเร็วกว่า TCP หากเกิดการสูญหายของ ICP-message ก็หมายความว่าเกิดความหนาแน่นหรือความเสียหายของเครือข่ายที่ใช้คิดต่อไปยังแคชที่อยู่ใกล้เคียง ในกรณีนี้จึงไม่ควรเลือกแคชตัวนี้ในรับออบเจกต์ที่ต้องการ นอกจากนี้ TCP ยังมี Overhead ที่ใหญ่กว่า UDP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบ ICP message

ICP Message จะประกอบด้วย 2 ส่วนคือ Header ขนาด 20 Octet และ Payload ซึ่งมีขนาดเปลี่ยนแปลงได้ดังรูปที่ 2.12 [4]



รูปที่ 2.12 รายละเอียดรูปแบบของ ICP Message

Opcode: เป็นส่วนที่ใช้บอกประเภทของ ICP-message โดยมีค่า Opcode ที่ใช้ทั่วไปดังนี้คือ ICP_QUERY, ICP_MISS, ICP_HIT เป็นต้น

Version: เป็นส่วนที่ใช้ระบุ version number ของ ICP protocol เพื่อใช้ตรวจสอบ โดยในปัจจุบันมีใช้งาน Version 2 และ 3

Message Length: เป็นส่วนที่ใช้บอกความยาวทั้งหมดของ ICP-message โดยจะมีความยาวไม่เกิน 16384 Octets

Request number: เป็นส่วนที่ใช้ Identify เมื่อมีการตอบ Query ค่านี้จะถูกเก็บลงใน Reply-message ด้วย

Options: มีความยาว 4 Octet เป็น Option flag ที่จะร่วมกับ Opcode เพื่อจุดประสงค์ต่าง ๆ เช่น ICP_FLAG_HIT_OBJ จะใช้ร่วมกับ ICP_QUERY เพื่อยินยอมให้ตอบกลับด้วยออบเจกต์เลย หากขนาดของออบเจกต์ไม่เกินความยาวของคำตอบ (Reply)

Options Data: มีขนาด 4 octet ใช้รองรับการทำงานของ Optional feature ซึ่ง ICP-feature อาจจะต้องใช้ข้อมูลในฟิลด์นี้

Sender Host Address: เป็นส่วนที่ใช้เก็บ IPv4 Address ของ Host ที่ส่ง ICP-Message แต่อย่างไรก็ตามก็มี Address ที่มีพร้อมกับ Socket API อยู่แล้วซึ่งเป็นการซ้ำซ้อน ทำให้ไม่เป็นที่นิยมในการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น

Payload: เป็นส่วนของเนื้อหาซึ่งมีขนาดไม่แน่นอนโดยจะขึ้นกับ Opcode แต่โดยปรกติแล้วจะเป็นค่า Null-terminated URL String

2.3 ไฟล์ Squid.conf กับพารามิเตอร์ที่มีผลต่อการทำงานของพร็อกซีเซิร์ฟเวอร์

ไฟล์ Squid.conf เป็นไฟล์สำคัญที่ใช้เก็บค่าพารามิเตอร์ต่าง ๆ ที่ Squid ต้องใช้ในการทำงาน มีลักษณะเป็นไฟล์ตัวอักษรที่สามารถเข้าไปแก้ไขได้ด้วยโปรแกรมอิดิเตอร์ทั่วไป โดยมีพารามิเตอร์ที่สำคัญและมีผลต่อการทำานในส่วนที่เกี่ยวข้องกับงานวิจัยดังต่อไปนี้

1. http_port เป็นการระบุหมายเลขพอร์ตให้ Squid ใช้ในการรอรับการร้องขอจาก HTTP ไคลเอนท์ ซึ่งการระบุนั้นสามารถทำได้ 3 แบบคือ 1 ระบุหมายเลขพอร์ตอย่างเดียว 2 ระบุชื่อเครื่องกับพอร์ต และ 3 ระบุ IP Address ของเครื่องกับพอร์ต ดังแสดงในตัวอย่างตามลำดับ

http_port *3218*

http_port *proxy.kmitl.ac.th:3128*

http_port *161.246.1.1:3128*

2. icp_port เป็นการระบุหมายเลขพอร์ตให้ Squid ใช้ในการรับ-ส่ง ICP-Query กับเครื่องพร็อกซีเซิร์ฟเวอร์อื่น โดยสามารถทำการระบุได้ทั้ง 3 แบบเช่นเดียวกับการระบุค่า http_port ดังที่แสดงไว้ข้างต้น

3. cache_peer เป็นการระบุรายชื่อของพร็อกซีเซอร์เวอร์เครื่องอื่นที่จะกำหนดให้ทำงานร่วมกับ Squid โดยในการตั้งค่า cache_peer นี้จะมีพารามิเตอร์อื่นนอกจากชื่อของ พร็อกซีเซอร์อีกดังรูปแบบต่อไปนี้

cache_peer *hostname type http_port proxy_port options*

โดยที่

Hostname หมายถึง ชื่อเครื่องพร็อกซีเซิร์ฟเวอร์ที่กำหนด

Type หมายถึง รูปแบบความสัมพันธ์ระหว่าง Squid กับพร็อกซีเซิร์ฟเวอร์ที่กำหนด โดยแบ่งเป็นความสัมพันธ์แบบ Parent หรือ Child

proxy_port หมายถึง หมายเลขพอร์ตของพร็อกซีเซิร์ฟเวอร์ที่กำหนดใช้รอรับการร้องขอ ซึ่งจะต้องเป็นหมายเลขเดียวกับ http_port ที่กำหนดไว้บนพร็อกซีเซิร์ฟเวอร์ที่กำหนดด้วย

icp_port หมายถึง หมายเลขพอร์ตของพร็อกซีเซิร์ฟเวอร์ที่กำหนดจะใช้รอรับการร้องขอ ICP-Query ซึ่งจะต้องเป็นหมายเลขเดียวกับ icp_port ที่กำหนดไว้บนพร็อกซีเซิร์ฟเวอร์ที่กำหนดด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

options จะเป็นค่าที่กำหนดรายละเอียดปลีกย่อยของการทำงานร่วมกับพร็อกซีเซิร์ฟเวอร์ที่กำหนด ซึ่งมีรายละเอียดดังนี้

- proxy-only เป็นการกำหนดให้มีการร้องขอออบเจกต์ที่ต้องการไปยังพร็อกซีเซอร์ที่กำหนดไว้แต่ไม่ต้องทำสำเนาออบเจกต์ที่ได้รับกลับมาไว้ที่ตัวเอง
- weight=n โดย n มีค่าเป็นเลขจำนวนเต็ม การกำหนดค่านี้ให้กับแต่ละเซิร์ฟเวอร์จะเป็นการบอกถึงว่าต้องการในการใช้งานเซิร์ฟเวอร์ว่ามากน้อยเพียงใด โดยที่ Squid จะเลือกใช้งานเซิร์ฟเวอร์ที่มีค่านี้มากที่สุดก่อน ตามปรกติถ้าไม่มีการกำหนดเป็นอย่างอื่นค่า weight จะมีค่าเท่ากับ 1
- no-query เป็นการกำหนดว่าไม่ต้องทำการส่ง ICP-Query กับพร็อกซีเซิร์ฟเวอร์นี้
- default เป็นการกำหนดให้พร็อกซีเซิร์ฟเวอร์นี้เป็นเครื่องที่ Squid จะส่งการร้องขอออบเจกต์ไปเมื่อไม่พบออบเจกต์ที่แคชของตัวเอง
- round-robin เป็นการหนดให้พร็อกซีเซิร์ฟเวอร์นี้อยู่ในรายชื่อของกลุ่มเครื่องที่จะมีการแบ่งภาระงานแบบ round-robin
- closest-only เป็นการกำหนดให้ใช้งานพร็อกซีนี้เมื่อไม่มีเซิร์ฟเวอร์เครื่องอื่นตอบกลับมาจาก ICP ว่าไม่พบออบเจกต์ที่ร้องขอไป
- login=user:password ใช้ในการส่ง Username และ Password ไปยังพร็อกซีเซิร์ฟเวอร์ที่กำหนด

ตัวอย่างการกำหนดค่า cache_peer ให้กับเครื่อง proxy1 ทำหน้าที่เป็น parent ที่ใช้ proxy_port เท่ากับ 8080 และ icp_port เท่ากับ 3130 โดยให้มี weight เท่ากับ 30 และไม่ต้องทำการสำเนาข้อมูลที่ได้รับจากเครื่อง proxy 1 นี้ สามารถเขียนแสดงได้ดังนี้

cache_peer *proxy1 parent 8080 3130 weight=30 proxy-only*

4. cache_mem เป็นการกำหนดขนาดของหน่วยความจำหลักที่จะนำมาใช้เป็นเมโมรีแคช มีหน่วยเป็นเมกะไบต์ (Mbytes) เช่นกำหนดให้เครื่องพร็อกซีเซิร์ฟเวอร์ใช้หน่วยความจำ 64 เมกะไบต์สำหรับใช้เป็นพื้นที่ของเมโมรีแคช สามารถเขียนได้ดังนี้

cache_mem *64 MB*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. cache_dir เป็นการระบุตำแหน่งบนดิสก์และขนาดที่จะให้ Squid นำมาใช้เป็นดิสก์แคช โดยมีรูปแบบในการระบุค่าดังนี้คือ **cache_dir** *type path size L1 L2* โดยที่

type คือ ประเภทของระบบดิสก์ที่ใช้ ซึ่งถ้าเป็นระบบปฏิบัติการ Unix ก็จะมีค่าเป็น ufs แต่ถ้าระบบดิสก์ที่ใช้เป็นแบบ Asynchronous I/O ก็จะมีค่าเป็น asyncufs

path คือ ตำแหน่งบนดิสก์ที่จะให้ Squid นำมาใช้ทำดิสก์แคช

size คือ ขนาดของดิสก์แคชมีหน่วยเป็นเมกะไบต์

L1 และ L2 คือ จำนวนของไดเรกทอรีย่อยในลำดับที่ 1 และ 2 นับจาก path ลงไป

ตัวอย่างการกำหนดค่า cache_dir โดยให้ Squid ทำดิสก์แคช ที่ไดเรกทอรี /cache/data โดยแบ่งไดเรกทอรีย่อยระดับที่ 1 และ 2 เท่ากับ 64 และ 128 ไดเรกทอรีย่อยตามลำดับและกำหนดขนาดของดิสก์แคชไว้ที่ 10 กิกะไบต์ (GigaBytes)

cache_dir *ufs /cache/data 10240 64 128*

6. cache_swap_low และ cache_swap_high เป็นการกำหนดขอบเขตในการลบออบเจกต์ออกจากพื้นที่เก็บข้อมูลของแคช โดยค่าทั้งสองจะมีหน่วยเป็นเปอร์เซ็นต์ โดย cache_swap_high จะใหม่ถึงขอบเขตที่เมื่อใดก็ตามที่มีออบเจกต์ที่จัดเก็บในแคชคิดเป็นเปอร์เซ็นต์เท่ากับค่า cache_swap_high Squid ก็จะทำการลบออบเจกต์บางส่วนออกด้วยกระบวนการในการแทนที่ออบเจกต์ จนออบเจกต์ที่เหลืออยู่มีการใช้พื้นที่ในการเก็บคิดเป็นเปอร์เซ็นต์ในแคชต่ำกว่าค่า cache_swap_in เช่น

cache_swap_low *80*

cache_swap_high *90*

7. maximum_object_size คือการกำหนดขนาดสูงสุดของออบเจกต์ที่อนุญาตให้เก็บไว้ในดิสก์แคชได้โดยกำหนดในหน่วยกิโลไบต์ ตัวอย่างเช่นกำหนดให้ขนาดของออบเจกต์ที่ใหญ่ที่สุดที่จะยอมให้เก็บในดิสก์แคชมีค่าเท่ากับ 4 เมกะไบต์ จะต้องกำหนดว่า

maximun_object_size *4096 KB*

8. cache_access_log คือการกำหนดไดเรกทอรีสำหรับใช้ในการเก็บล็อกไฟล์ (access.log) ซึ่งเป็นผลมากจาการทำงานของ Squid โดยทำการระบุไดเรกทอรีปลายทางที่กำหนดให้เป็นที่จัดเก็บไฟล์ พร้อมทั้งกำหนดชื่อไฟลที่จะจัดเก็บ ตัวอย่างการกำหนดให้ Squid เก็บล็อกไฟล์ชื่อ access.log ไว้ที่ไดเรกทอรี /usr/local/squid/logs/

cache_access_log */usr/local/squid/logs/access.log*

เอกสาร...การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้าม...ารทุกครั้งที่มีการนำไปใช้

9. reference_age คือ ค่าอายุอ้างอิงที่ Squid จะใช้ในการเปรียบเทียบว่าออบเจกต์แต่ละออบเจกต์ที่ถูกจัดเก็บอยู่ในแคชนั้นหมดอายุหรือยังเพื่อที่จะทำการลบออบเจกต์ที่มีอายุเกินค่าที่กำหนดนี้ออกจากพื้นที่ของดิสก์แคชในกรณีที่มีความต้องการใช้พื้นที่นั้นในการเก็บออบเจกต์ใหม่ที่เข้ามา โดยหน่วยของเวลาที่สามารถกำหนดนั้นจะมีได้ตั้งแต่ ชั่วโมง วัน สัปดาห์ และ เดือน ตัวอย่างเช่น กำหนดให้ค่าอายุอ้างอิงเท่ากับ 2 สัปดาห์จะได้ว่า

reference_age *2 weeks*

10. replacement_policy คือ การกำหนดให้ Squid ใช้กระบวนการในการแทนที่ออบเจกต์ใหม่ลงในแคชโดยจะมีวิธีเลือกลบออบเจกต์ออกจากแคชที่แตกต่างกัน เช่น Least-Recently Used (LRU), Greedy-Dual Size Frequency (GDSF) และ Least Frequently Used with Dynamic Aging (LFUDA) ตัวอย่างเช่นการกำหนดให้ Squid ใช้ replacement policy แบบ LFUDA

replacement_policy *LFUDA*

2.4 ล็อกไฟล์

เป็นไฟล์ตัวอักษร (Text file) ที่เก็บรายละเอียดของผลการทำงานของ Squid ที่เกิดจากการร้องขอออบเจกต์จากผู้ใช้ในแต่ละการร้องขอ โดยในหนึ่งบรรทัดของล็อกไฟล์นี้ก็คือผลของการร้องขอที่มีเข้ามายัง Squid ซึ่งใน 1 บรรทัดนั้นจะแบ่งออกเป็น 10 ฟิลด์ [7] ดังแสดงในตารางที่ 2.1

ตารางที่ 2.1 ฟิลด์ต่าง ๆ ของล็อกไฟล์ที่เกิดจากการประมวลผลการร้องขอของ Squid

Timestamp	Elapsed	Client-address	Log-Tag/ HTTP-Code	Size	Request Method	URL	rfc931	Hierarchy/ Hostname	Content-Type

1. Timestamp

เป็นเวลาในรูปแบบของระบบปฏิบัติการแบบ Unix โดยเริ่มนับเวลาทีละ 1 วินาที ตั้งแต่วันที่ 1 มกราคม ค.ศ.1970 (Unix Time) โดยการบันทึกเวลาลงล็อกไฟล์จะกระทำก็ต่อเมื่อการเชื่อมต่อกับไคลเอนท์ถูกปิดลง

2. Elapsed Time

คือเวลาที่ใช้ในการประมวลผลการร้องขอ ตั้งแต่เริ่มการเชื่อมต่อจนสิ้นสุดการเชื่อมต่อ มีหน่วยเป็นมิลลิวินาที (Millisecond, ms)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Client Address

คือ IP Address ของไคลเอนท์ที่ร้องขอออบเจกต์ ในกรณีที่เปิดการใช้งาน log_fqdn ใน file squid.conf ก็จะเก็บ Full Qualify Domain Name แทน

4. Log Tag / HTTP Code

เป็นส่วนที่ใช้อธิบายว่าการร้องขอที่เข้ามามีผลจากการทำงานเป็นอย่างไร เช่น พบ หรือ ไม่พบออบเจกต์ที่ร้องขอ ซึ่งจะอธิบายต่อไป ส่วน HTTP code คือข้อมูลในส่วนของ HTTP header ที่ตอบกลับมาจากเซิร์ฟเวอร์ที่พร็อกซีไปร้องขอออบเจกต์มา ซึ่งจะได้อธิบายเพิ่มเติมในส่วนถัดไป

5. Size

ขนาดของออบเจกต์ที่ถูกร้องขอ มีหน่วยเป็นไบต์

6. Request Method

จะเก็บวิธีการกระทำต่อออบเจกต์ที่ร้องขอเข้ามา เช่น GET (นำมา) POST (ส่งออกไป) หรือ ICP_Query

7. URL

เก็บ URL ของออบเจกต์ที่ถูกร้องขอ

8. Ident

ในกรณีที่มีการใช้ความสามารถในการทำ Ident (โดยปรับตั้งค่า ident_lookup ที่ไฟล์ squid.conf)

9. Hierarchy Data / Hostname

แสดงรายละเอียดว่า ออบเจกต์นั้นถูกนำมาจากไหนและอย่างไร เช่น นำมาโดยตรงจากเซิร์ฟเวอร์ต้นทาง หรือ พบที่ Parent ของตัวมัน ซึ่งจะได้อธิบายเกี่ยวกับ Hierarchy Data เพิ่มเติมในส่วนถัดไป

10. Content Type

เก็บข้อมูลประเภทของออบเจกต์ที่ได้รับกลับมาจากเซิร์ฟเวอร์ต้นทาง เช่น เป็นข้อมูล Text ข้อมูลรูปภาพกราฟฟิก หรือข้อมูลมัลติมีเดีย เป็นต้น

รายละเอียดของ Log Tag

Log tag ที่ขึ้นต้นด้วย TCP_ จะหมายถึงการเชื่อมต่อด้วยโพรโตคอล HTTP จากไคลเอนท์เข้ามายัง Squid (โดยเข้ามาที่ พอร์ต 3128 ถ้าไม่มีการกำหนดเป็นอย่างอื่นในค่า tcp_port ในไฟล์squid.conf)

- TCP_HIT หมายถึง ออบเจกต์ที่ไคลเอนท์ร้องขอพบอยู่ในแคชและสามารถส่งให้กับไคลเอนท์ได้
- TCP_MEM_HIT หมายถึง พบออบเจกต์ที่ร้องขออยู่ใน เมโมรีแคช
- TCP_MISS หมายถึง ออบเจกต์ที่ไคลเอนท์ร้องขอไม่พบอยู่ในแคช

เอกสารนี้เป็นเอกสารที่สงวน... ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้ง... เนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- TCP_REFRESH_HIT หมายถึง ออบเจกต์ที่ไคลเอนท์ร้องขอพบอยู่ในแคชแต่หมดอายุ จากนั้น Squid ได้ทำการส่งการร้องขอแบบ If-Modify-Since ไปยังเซิร์ฟเวอร์ต้นทางและได้รับคำตอบกลับมาทาง HTTP reply ว่า "304 Not modified" ซึ่งหมายถึงว่าออบเจกต์นั้นยังไม่มีการเปลี่ยนแปลง (ขนาดและ/หรือวันที่สร้าง) จากนั้น squid จึงส่งออบเจกต์ที่เก็บอยู่ในแคชไปให้ไคลเอนท์พร้อมทั้งเริ่มนับเวลาที่ออบเจกต์ถูกเก็บในแคชใหม่
- TCP_REF_FAIL_HIT หมายถึง ออบเจกต์ที่ไคลเอนท์ร้องขอพบอยู่ในแคชแต่หมดอายุ จากนั้น Squid ได้ทำการส่งการร้องขอแบบ If-Modify-Since ไปยังเซิร์ฟเวอร์ต้นทางแต่ไม่ได้รับการตอบกลับมา Squid จึงทำการส่งออบเจกต์ที่พบในแคชแต่หมดอายุนั้นไปให้ไคลเอนท์
- TCP_REFRESH_MISS หมายถึง ออบเจกต์ที่ไคลเอนท์ร้องขอพบอยู่ในแคชแต่หมดอายุ จากนั้น Squid ได้ทำการส่งการร้องขอแบบ If-Modify-Since ไปยังเซิร์ฟเวอร์ต้นทางและได้รับออบเจกต์ใหม่กลับมา
- TCP_CLIENT_REFRESH หมายถึง ไคลเอนท์ระบุการร้องขอแบบไม่ใช้แคช (คือขอให้ Squid ไปร้องขอออบเจกต์นั้นโดยตรงจากเซิร์ฟเวอร์ต้นทาง)
- TCP_IMS_HIT หมายถึง ไคลเอนท์ส่งการร้องขอแบบ If-Modified-Since เข้ามาแต่ปรากฏว่าพบออบเจกต์ในแคชและออบเจกต์ยังไม่หมดอายุ Squid จึงทำการส่งออบเจกต์ที่พบในแคชนั้นให้กับไคลเอนท์
- TCP_IMS_MISS หมายถึง ไคลเอนท์ส่งการร้องขอแบบ If-Modified-Since เข้ามาและปรากฏว่าพบออบเจกต์ในแคชแต่ออบเจกต์หมดอายุไปแล้ว Squid จึงต้องทำการร้องขอออบเจกต์นั้นใหม่
- TCP_SWAPFAIL หมายถึง ออบเจกต์ที่ร้องขอถูกลงทะเบียนไว้ในรายชื่อของออบเจกต์ที่ถูกจัดเก็บอยู่ในดิสก์แคช แต่ไม่สามารถเข้าถึงได้ กรณีนี้อาจจะเกิดจากการหยุดการทำงานอย่างผิดปรกติของ Squid process ทำให้เกิดความผิดพลาดในการจัดเก็บออบเจกต์ในดิสก์แคช
- TCP_DENIED หมายถึง การร้องขอจากไคลเอนท์นี้ถูกปฏิเสธ ซึ่งเป็นผลมาจากการทำงานในส่วนของ Access Control List

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Log tag ที่ขึ้นต้นด้วย UDP_ จะหมายถึงการเชื่อมต่อด้วยโพรโตคอล ICP จากไคลเอนท์เข้ามายัง Squid (โดยเข้ามาที่ พอร์ต 3130 ถ้าไม่มีการกำหนดเป็นอย่างอื่นในค่า icp_port ในไฟล์ squid.conf)

- UDP_HIT หมายถึง ออบเจกต์ที่ไคลเอนท์ร้องขอพบอยู่ในแคชของพร็อกซีเซิร์ฟเวอร์ปลายทาง (ที่ Squid ส่ง ICP-Query ไปร้องขอ) และสามารถส่งให้กับไคลเอนท์ได้
- UDP_HIT_OBJ หมายถึง ออบเจกต์ที่ไคลเอนท์ร้องขอถูกพบเหมือนกับกรณีของ UDP_HIT แต่ตัวออบเจกต์ที่ต้องการนั้นมีขนาดเล็กพอที่จะส่งกลับมาพร้อมกับคำตอบว่าพบออบเจกต์นั้นเลย (ส่งกลับมาในส่วนของ TCP-Reply)
- UDP_MISS หมายถึง ไม่พบออบเจกต์ที่ไคลเอนท์ร้องขออยู่ในแคชของพร็อกซีเซิร์ฟเวอร์ปลายทาง (ที่ Squid ส่ง ICP-Query ไปร้องขอ)
- UDP_DENIED หมายถึง การร้องขอจากไคลเอนท์นี้ถูกปฏิเสธ ซึ่งเป็นผลมาจากการทำงานในส่วนของ Access Control List
- UDP_INVALID หมายถึง การร้องขอจากไคลเอนท์ที่ไม่ถูกต้อง
- UDP_RELOADING หมายถึง การร้องขอถูกปฏิเสธเนื่องจาก Squid กำลังเริ่มต้นการทำงานของตัวอยู่และยังไม่เสร็จสิ้นกระบวนการเรียกข้อมูลขึ้นมาเพื่อใช้ในการทำงาน
- Log tag ที่ขึ้นต้นด้วย "ERR_" หมายถึง การร้องขอแบบ HTTP ที่ผิดพลาดไม่สามารถตีความหรือทำงานได้อย่างถูกต้อง

รายละเอียดของ Hierarchy Data

ในส่วนของ Hierarchy Data นี้จะเป็นข้อมูลที่จะบ่งบอกให้ทราบถึงลำดับขั้นและผลของการร้องขอออบเจกต์จากพร็อกซีเซิร์ฟเวอร์ว่ามีการร้องขอออบเจกต์ที่ไม่พบในแคชของตัวอย่างผ่านทางไหนอย่างไรบ้างและผลของการร้องขอนั้นปรากฏผลว่าเป็นอย่างไร ซึ่งจะมี Hierachy Data ดังแสดงต่อไปนี้

- DIRECT หมายความว่า Squid นำออบเจกต์ที่ถูกร้องขอนั้นมาจากเซิร์ฟเวอร์ต้นทางตามที่ระบุไว้ใน URL โดยตรงเลย
- FIREWALL_IP_DIRECT หมายความว่า Squid นำออบเจกต์ที่ถูกร้องขอนั้นมาจากเซิร์ฟเวอร์ต้นทางตามที่ระบุไว้ใน URL โดยตรงเนื่องจากเซิร์ฟเวอร์ต้นทางนั้นอยู่ภายในเครือข่ายหลังไฟร์วอลตัวเดียวกับ Squid

เอกสารนี้เป็นเอกสารที่สงวน ... านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้ง ... ใช้

- FIRST_PARENT_MISS หมายความว่า Squid นำออบเจกต์ที่ถูกร้องขอนั้นมาจาก Parent ที่ติดต่อได้เร็วที่สุด
- FIRST_UP_PARENT หมายความว่า Squid นำออบเจกต์ที่ถูกร้องขอนั้นมาจาก Parent ที่ว่างในรายชื่อของ parent ทั้งหมดที่มี
- LOCAL_IP_DIRECT หมายความว่า Squid นำออบเจกต์ที่ถูกร้องขอนั้นมาจากเซิร์ฟเวอร์ต้นทางตามที่ระบุไว้ใน URL โดยตรงเนื่องจากเซิร์ฟเวอร์ต้นทางนั้นอยู่ในรายชื่อของโฮสต์ที่ระบุไว้ใน local_ip ในไฟล์ squid.conf
- SIBLING_HIT หมายความว่า Squid นำออบเจกต์ที่ถูกร้องขอนั้นมาจากพร็อกซีเซิร์ฟเวอร์ที่อยู่ในระดับเดียวกัน (Sibling)
- NO_DIRECT_FAIL หมายความว่า Squid ไม่สามารถนำออบเจกต์ที่ถูกร้องขอมาได้เนื่องจากไม่สามารถผ่านไฟลวอลล์หรือไม่มีพร็อกซีเซิร์ฟเวอร์ที่เป็น Parent ว่างในการรับการร้องขอจาก Squid
- NO_PARENT_DIRECT หมายความว่า Squid นำออบเจกต์ที่ถูกร้องขอนั้นมาจากเซิร์ฟเวอร์ต้นทางอันเนื่องมากจากไม่มีรายชื่อของพร็อกซีเซิร์ฟเวอร์ที่เป็น parent ที่กำหนดไว้สำหรับออบเจกต์ที่มีการร้องขอเข้ามา
- PARENT_HIT หมายความว่า Squid นำออบเจกต์ที่ถูกร้องขอนั้นมาจากพร็อกซีเซิร์ฟเวอร์ที่เป็น Parent
- SINGLE_PARENT หมายความว่าออบเจกต์นี้เป็น URL ที่กำหนดไว้ว่าต้องถูกร้องขอจาก Parent ที่ระบุไว้เท่านั้น
- SOURCE_FASTEST หมายความว่า Squid นำออบเจกต์ที่ถูกร้องขอนั้นมาจากเซิร์ฟเวอร์ต้นทางเนื่องมากจากเครื่องเซิร์ฟเวอร์ต้นทางตอบกลับมาก่อนพร็อกซีเซริฟเวอร์ตัวอื่น
- PARENT_UDP_HIT_OBJ หมายความว่าออบเจกต์ที่ถูกร้องขอนั้นได้รับกลับมาในการตอบรับ UDP_HIT_OBJ reply จาก Parent เลย
- SIBLING_UDP_HIT_OBJ หมายความว่าออบเจกต์ที่ถูกร้องขอนั้นได้รับกลับมาในการตอบรับ UDP_HIT_OBJ reply จาก Sibling เลย
- PASSTHROUGH_PARENT หมายความว่าออบเจกต์ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์ที่ระบุไว้ใน passthrough_proxy ในไฟล์ squid.conf
- SSL_PARENT_MISS หมายความว่าออบเจกต์ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์ที่ระบุไว้ใน ssl_proxy ในไฟล์ squid.conf.

- DEFAULT_PARENT หมายความว่าออบเจกต์ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์ที่กำหนดไว้เป็น default ใน cache_peer ในไฟล์ squid.conf โดยที่ไม่ได้มีการส่ง ICP-Message ออกไปเพื่อหาพร็อกซีที่มีการตอบสนองเร็วที่สุด
- ROUNDROBIN_PARENT หมายความว่าออบเจกต์ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์ที่กำหนดไว้เป็น default ใน cache_peer ในไฟล์ squid.conf และเป็นพร็อกซีเซิร์ฟเวอร์ที่มีการใช้งานน้อยที่สุดจากกระบวนการแบ่งภาระงานแบบ Round-robin โดยที่ไม่ได้มีการส่ง ICP-Message ออกไปเพื่อหาพร็อกซีที่มีการตอบสนองเร็วที่สุด
- CLOSEST_PARENT_MISS หมายความว่าออบเจกต์ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์ที่ทีค่า Round-trip time ไปยังเซิร์ฟเวอร์ต้นทางต่ำที่สุด ซึ่งกรณีนี้จะเกิดขึ้นก็ต่อเมื่อมีการเปิดใช้งาน query_icmp ในไฟล์ squid.conf
- CLOSEST_DIRECT หมายความว่าออบเจกต์ถูกร้องขอไปยังเซิร์ฟเวอร์ต้นทางโดยตรงเนื่องจากตัว Squid เองเป็นผู้ที่มีค่า Round-trip time ต่ำที่สุด

2.5 สรุป

ในปัจจุบันมีผู้พัฒนาโปรแกรมพร็อกซีเซิร์ฟเวอร์เป็นจำนวนมากเพื่อใช้ประโยชน์ในการเพิ่มประสิทธิภาพในการสืบค้นข้อมูลจากอินเทอร์เน็ต ซึ่ง Squid เป็นพร็อกซีเซิร์ฟเวอร์ที่ได้รับความนิยมเป็นอย่างมากและได้ถูกนำมาใช้งานเป็นพร็อกซีเซิร์ฟเวอร์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง โดยการทำงานของพร็อกซีนั้นสามารถทำงานได้อย่างอิสระหรือกำหนดให้ทำงานร่วมกันก็ได้และผลของการทำงานในการรับการร้องของนั้นจะถูกบันทึกเก็บลงในล็อกไฟล์ ซึ่งเราสามารถนำล็อกไฟล์นี้ไปใช้ประโยชน์ต่อไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การหาค่าพารามิเตอร์ที่สำคัญและเกี่ยวข้องกับการปรับแต่งพร็อกซีเซิร์ฟเวอร์

ในบทนี้จะทำการอธิบายถึงเสนอวิธีการและอัลกอริธึมในการหาค่าขนาดและอายุของออบเจกต์ที่ถูกร้องขอผ่าน พร็อกซีเซิร์ฟเวอร์ โดยอาศัยข้อมูลที่อยู่ในล็อกไฟล์ซึ่งเป็นผลจากการทำงานของพร็อกซีเซิร์ฟเวอร์ นอกจากนี้ยังได้แสดงการวิเคราะห์ความผิดพลาดของอัลกอริธึมดังกล่าว ตลอดจนทำการทดลองใช้อัลกอริธึมนี้หาค่าขนาดและอายุของออบเจกต์จริงโดยใช้ล็อกไฟล์จากเครื่องพร็อกซีเซิร์ฟเวอร์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบังอีกด้วย

3.1 พารามิเตอร์ที่สำคัญและวัดได้จากล็อกไฟล์

เมื่อพร็อกซีเซิร์ฟเวอร์ทำงานตามการร้องขอจากไคลเอนต์แต่ละรายการเสร็จก็จะทำการบันทึกผลการทำงานของการร้องขอนั้น ลงสู่ล็อกไฟล์เพื่อแสดงรายละเอียดต่าง ๆ ซึ่งในงานวิจัยนี้จะใช้ไฟล์ชื่อ access.log ของ Squid เวอร์ชัน 2.2 [6] โดยในไฟล์ access.log 1 บรรทัดจะประกอบด้วย 10 ฟิลด์ดังที่ได้กล่าวถึงไปแล้วในบทที่ 2 ซึ่งรูปแบบของล็อกไฟล์ของ Squid ตั้งแต่เวอร์ชั่น 2 เป็นต้นมาจะใช้รูปแบบและการจัดเรียงฟิลด์เหมือนกันทั้งหมด

ในที่นี้จะขออธิบายความหมายเฉพาะฟิลด์ที่เกี่ยวข้องและนำมาใช้ในงานวิจัยเพื่อทำการหาอายุและขนาดของออบเจกต์เท่านั้น ได้แก่

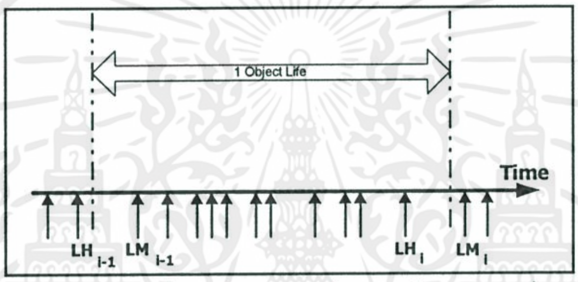
- Timestamp คือ ค่าเวลาปัจจุบันในรูปแบบของ Unix time ที่ Squid จะบันทึกไว้เมื่อเสร็จสิ้นการร้องขอนั้น
- Log-Tag/HTTP-Code คือ ผลการทำงานของ Squid ที่มีต่อการร้องขอนั้น/ผลการทำงานจากโพรโตคอล HTTP
- Size คือ ขนาดของออบเจกต์ที่ไคลเอนต์ร้องขอมีหน่วยเป็นไบต์
- Req.-Method คือ วิธีการของการร้องขอเพื่อจะกระทำกับออบเจกต์ เช่น GET, POST เป็นต้น
- URL คือ URL ที่ไคลเอนต์ร้องขอ ซึ่งในที่นี้จะใช้เป็นตัวระบุชื่อเพื่อแยกแยะแต่ละออบเจกต์ที่แตกต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยในแต่ละบรรทัดของไฟล์ access.log นี้จะเก็บบันทึกข้อมูลที่เกี่ยวกับออบเจกต์ต่างๆ เรียงตามลำดับเวลาก่อนหลังทำให้สามารถนำไฟล์นี้มาใช้ทำการประมวลผลเพื่อหาอายุและขนาดของออบเจกต์ได้ตามลำดับเหตุการณ์ที่เป็นจริง

3.2 อัลกอริธึมในการวัดค่าอายุและขนาดของออบเจกต์

การหาค่าอายุและขนาดของออบเจกต์ (Object) ในอินเทอร์เน็ตที่ถูกร้องขอจากผู้ใช้โดยผ่านพร็อกซีจะอาศัยข้อมูลจากล็อกไฟล์ของโปรแกรมพร็อกซีเซิร์ฟเวอร์ ข้อมูลแต่ละบรรทัดของล็อกไฟล์คือรายละเอียดของข้อมูลที่แต่ละไคลเอนต์ร้องขอเข้ามายังตัวพร็อกซีเซิร์ฟเวอร์ เมื่อนำข้อมูลจากล็อกไฟล์มาเขียนแทนด้วยลูกศรบนแกนเวลา ก็จะได้แผนภาพดังแสดงในรูปที่ 3.1



รูปที่ 3.1 หลักการหาอายุของออบเจกต์จากข้อมูลการร้องขอข้อมูลในพร็อกซีเซิร์ฟเวอร์

จากรูปที่ 3.1 มีการร้องขอออบเจกต์เข้ามาทางซ้ายสุดของแกนเวลาซึ่งอาจจะเป็นการร้องขอแรกที่บันทึกไว้ในล็อกไฟล์และต่อมาเมื่อมีการร้องขอออบเจกต์ที่เราสนใจนี้เข้ามาอีกและพร็อกซีเซิร์ฟเวอร์พบ ออบเจกต์นั้นในแคชของและได้ทำการตรวจสอบแล้วว่าออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตนั้นยังไม่มีการเปลี่ยนแปลงขนาดและ/หรือ วันที่ออบเจกต์นั้นถูกสร้าง จึงได้ทำการนำออบเจกต์นั้นที่เก็บไว้ในแคชส่งไปให้ไคลเอนท์ที่ร้องขอมา เราจะจัดให้การร้องขอดังกล่าวนี้อยู่ในประเภทของการร้องขอแบบ Life_Hit (LH) จากนั้นการร้องขอครั้งถัดไปดังแสดงในรูปเมื่อมีการร้องขอออบเจกต์นี้เข้ามาอีกแต่ออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตมีการเปลี่ยนแปลงพร็อกซีเซิร์ฟเวอร์ก็จะต้องไปเอาออบเจกต์นี้มาเก็บลงในแคชแทนออบเจกต์ตัวเดิมที่มีอยู่ก่อนหน้านี้และทำการส่งออบเจกต์นั้นให้กับไคลเอนท์ ในกรณีนี้เราจะจัดให้การร้องขอดังกล่าวอยู่ในประเภทของการร้องขอแบบ Life_Miss (LM) ซึ่งในความเป็นจริงการร้องขอที่เข้ามายังพร็อกซีเซิร์ฟเวอร์นั้นอาจจะมีการร้องขอออบเจกต์เดียวกันจากหลาย ๆ ไคลเอนท์ในเวลาไล่เลี่ยกัน จึงมีโอกาสของการเกิดกรณีของ Life_Hit มากกว่า Life_Miss นอกจากนี้ยังอาจจะมีผลของการร้องขอที่ยังไม่สามารถระบุได้แน่ชัดว่าออบเจกต์มีการเปลี่ยนแปลงหรือไม่ซึ่งเราจะจัดให้อยู่ในประเภทของการร้องขอแบบ Life_Unknown (LU) ส่วนกรณีของการร้องขอที่ไม่สำเร็จหรือไม่มีผลต่อการหาอายุและขนาด

ของออบเจกต์จะจัดอยู่ในการร้องขอแบบ Life_Ignore (LI) ซึ่งเราจะไม่นำมาใช้ในการคำนวณหาอายุและขนาดของออบเจกต์

เราสามารถที่จะจัดแยกประเภทของการร้องขอแต่ละครั้งเพื่อนำไปใช้ในการคำนวนหาอายุของออบเจกต์ ได้จากข้อมูลผลของการร้องขอ (Request Type) ซึ่งเป็นข้อมูลในฟิลด์ที่ 4 ในส่วนของ Log-Tag ในแต่ละบรรทัดของไฟล์ access.log โดยได้จัดแยกประเภทของการร้องขอตามผลของการร้องขอดังแสดงในตารางที่ 3-1

ตารางที่ 3.1 ประเภทของ การร้องขอ ที่มีผลต่อการหาอายุของออบเจกต์

ประเภท	ประเภทของการร้องขอ (Request Type)	รายละเอียด
Life_Miss (LM)	TCP_REFRESH_MISS	คือการร้องขอที่เกิดเมื่อ Object มีการเปลี่ยนแปลงอายุหรือวันที่สร้าง
Life_Hit (LH)	TCP_REFRESH_HIT, TCP_IMS_HIT, TCP_IMS_MISS	เป็น การร้องขอที่เกิดเมื่อ Object ไม่มีการเปลี่ยนแปลงอายุหรือวันที่สร้าง
Life_Unknown (LU)	TCP_HIT, TCP_MISS, UDP_HIT, UDP_MISS, TCP_MEM_HIT, TCP_CLIENT_REFRESH_MISS	เป็นการร้องขอที่ยังไม่สามารถชี้ชัดได้ว่า Object มีการเปลี่ยนแปลงหรือไม่ จำเป็นต้องดูการเปลี่ยนแปลงขนาดและวันที่สร้างประกอบ
Life_Ignore (LI)	NONE, ERR_*, TCP_DENIDE, TCP_NEGATIVE_HIT, TCP_SWAPFAIL	เป็นการร้องขอที่ไม่เกี่ยวข้องกับการเปลี่ยนแปลง Object ซึ่งจะไม่นำมาประมวลผล

โดยที่

- Life_Miss (LM) หมายถึง ผลของการร้องขอที่เกิดขึ้นหลังจากออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตมีการเปลี่ยนแปลงขนาดหรือวันที่ออบเจกต์นั้น ๆ ถูกสร้างขึ้นมา (Created Date)
- Life_Hit (LH) หมายถึง ผลของการร้องขอที่เกิดขึ้นในขณะที่ออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตไม่มีการเปลี่ยนแปลงทั้งขนาดและวันที่ถูกสร้าง
- Life_Unknown (LU) หมายถึง ผลของการร้องขอที่ยังไม่สามารถชี้ชัดได้ว่าออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตมีการเปลี่ยนแปลงขนาดและ/หรือวันที่ถูกสร้างหรือไม่

- Life_Ignore (LI) หมายถึง ผลของการร้องขอที่ไม่เกี่ยวข้องกับการเปลี่ยนแปลงของออบเจกต์ ซึ่งจะไม่นำมาประมวลผลแต่อย่างใด

ประเภทของการร้องขอที่จะนำมาใช้ในการคำนวณหาอายุของออบเจกต์นั้นได้แก่ประเภท Life_Miss, Life_Hit และ Life_Unknown ซึ่งในสองประเภทแรกสามารถนำไปคำนวณหาอายุได้เลยโดยจะจัดให้มีความแม่นยำของอายุที่หาได้จากกรณีนี้อยู่ในระดับที่ 1 (Accuracy Level 1) ทั้งนี้จะกำหนดให้ค่าอายุจริงของออบเจกต์มีความแม่นยำอยู่ในระดับ 0 (Accuracy Level 0) ส่วนในกรณีของผลของการร้องขอแบบ Life_Unknown นั้นจะต้องมีการตรวจสอบก่อนว่าการร้องขอออบเจกต์ที่ทำให้เกิดกรณีของ Life_Unknown นั้นตัวออบเจกต์ที่ถูกร้องขอมีการเปลี่ยนแปลงขนาดหรือไม่ ถ้ามีการเปลี่ยนแปลงขนาดจึงจะถือว่าออบเจกต์นั้นมีการเปลี่ยนแปลงและจะอนุโลมให้การร้องขอนี้จัดอยู่ในประเภท Life_Miss เพื่อนำไปใช้ในการคำนวณต่อไปและจะจัดให้ความแม่นยำของค่าอายุที่หาได้จากกรณี Life_Unknown นี้อยู่ในระดับที่ 2 (Accuracy Level 2)

เราสามารถประมาณอายุของออบเจกต์ ในช่วงที่ i ($LifeCal_i$) ได้จากการนำเวลาของการเกิด Life_Miss ครั้งล่าสุด(LM_i) และ Life_Hit ครั้งล่าสุด(LH_i)มาหาค่าเฉลี่ยแล้วลบด้วยค่าเฉลี่ยของเวลาที่เกิดการเปลี่ยนแปลงของออบเจกต์ในช่วงก่อนหน้านั้น(LM_{i-1} , LH_{i-1}) ดังแสดงในสมการที่ 1

$$Lifecal_i = \frac{(LM_i + LH_i)}{2} - \frac{(LM_{i-1} + LH_{i-1})}{2} \qquad (3.1)$$

ในส่วนของขนาดของออบเจกต์นั้นเราสามารถหาได้จากฟิลด์ที่ 5 (Size) จากไฟล์ access.log โดยมีข้อแม้ว่าเราไม่ได้เก็บค่าขนาดของออบเจกต์ทุกๆ ค่าที่ปรากฏอยู่ในไฟล์นั้น แต่จะเก็บค่าขนาดของออบเจกต์ก็ต่อเมื่อพบว่าออบเจกต์นั้นๆ มีการเปลี่ยนแปลงเกิดขึ้นเท่านั้น (ในกรณีที่เกิดการร้องขอที่จัดอยู่ในประเภท Life_Miss) โดยจะนับจำนวนของอายุของ ออบเจกต์ไปด้วย

จากวิธีการคำนวณอายุของออบเจกต์ที่กล่าวมาข้างต้นสามารถสรุปออกมาเป็นอัลกอริธึมได้ดังนี้ ขั้นแรกเริ่มจากการจัดประเภทของการร้องขอออก ประเภทต่าง ๆ ตามผลของการร้องขอที่ปรากฏใน Log-Tag ของ Squid ซึ่งจะได้ผลออกมาเป็น 3 กรณีที่สามารถนำมาคำนวณหาอายุของออบเจกต์ดังนี้คือ

1. กรณี Life Hit (LH) ให้นำ URL ที่ถูกร้องขอนั้นไปหาในฐานข้อมูลว่ามี URL นั้นอยู่ในฐานข้อมูลหรือไม่ ถ้าไม่พบ แสดงว่าเป็น URL ใหม่ที่ไม่เคยมีการเก็บข้อมูลมาก่อน ให้ทำการเก็บบันทึกข้อมูล URL, เวลา ลงในฐานข้อมูล ถ้าพบว่ามี URL อยู่แล้วให้ทำการเปรียบเทียบว่าขนาดของ URL ปัจจุบันมีค่าเท่ากับขนาดของ URL ที่พบในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือไม่ ถ้าขนาดเท่ากัน ให้ทำการเพิ่มค่าจำนวนของการร้องขอที่มีเข้ามาในช่วงอายุนี้ (Count) และทำการเก็บค่าเวลาที่เกิดการร้องขอนี้ไว้เป็นเวลาล่าสุดที่มีการร้องขอออบเจกต์ เพื่อใช้ในการคำนวนเมื่อมีการร้องขออันถัดไปเข้ามา ถ้าพบว่าขนาดของ URL ที่พบไม่เท่ากับขนาดของ URL ปัจจุบันให้บันทึกลง Error log

2. กรณี Life Miss (LM) ให้นำ URL ที่ถูกร้องขอนั้นไปหาในฐานข้อมูลว่ามี URL นั้นอยู่ในฐานข้อมูลหรือไม่ ถ้าไม่พบแสดงว่าเป็น URL ใหม่ที่ไม่เคยมีการเก็บข้อมูลมาก่อน ให้ทำการเก็บบันทึกข้อมูล URL, เวลาที่เกิดการร้องขอนี้ซึ่งเป็นเวลาที่ถือว่าออบเจกต์นี้เกิดในช่วงอายุใหม่ (Birth) ลงในฐานข้อมูล ถ้าพบว่ามี URL อยู่แล้วให้ทำการเปรียบเทียบค่า เวลาที่ออบเจกต์เกิดในช่วงอายุล่าสุด

2.1 ถ้าเวลาที่ออบเจกต์เกิดในช่วงอายุล่าสุดมีค่าเท่ากับ 0 ซึ่งหมายความว่ายังไม่เคยมีการบันทึกการร้องขอที่ออบเจกต์มีการเกิดมาเลย ให้ทำการตรวจสอบเวลาในการร้องขอครั้งล่าสุดของออบเจกต์นั้นว่ามีค่าเท่ากับ 0 หรือไม่.ถ้าเวลาในการร้องขอครั้งล่าสุดของ ออบเจกต์นั้นเท่ากับ 0 (ไม่เคยมีการร้องขอออบเจกต์นี้มาก่อน) ให้กำหนดค่าเวลาในการเกิดของออบเจกต์นี้เท่ากับเวลาปัจจุบันที่มีการร้องขอนี้เข้า ถ้าเวลาในการร้องขอครั้งล่าสุดของออบเจกต์นั้นเท่าไม่กับ 0 ให้กำหนดค่าเวลาในการเกิดของออบเจกต์นี้จากค่าเฉลี่ยระหว่างเวลาปัจจุบันที่มีการร้องขอนี้กับเวลาในการร้องขอครั้งล่าสุดก่อนหน้าการร้องขอปัจจุบัน

จากนั้นให้ทำการกำหนดค่า การร้องขอครั้งล่าสุดของออบเจกต์นี้เท่ากับเวลาปัจจุบันและทำการบันทึกข้อมูล ของ URL นี้และปิด Record ปัจจุบัน (จบช่วงอายุ) จากนั้นให้ทำการเปิด Record ใหม่ในฐานข้อมูลในฐานะของออบเจกต์เกิดใหม่ (ช่วงอายุใหม่) โดยกำหนดให้ค่า เวลาในการเกิดของออบเจกต์ให้มีค่าเท่ากับที่คำนวณได้จากข้างต้น และกำหนดให้ค่าการร้องขอครั้งล่าสุดของออบเจกต์ในอายุใหม่นี้มีค่าเท่ากับเวลาปัจจุบันของการร้องขอออบเจกต์นี้ในช่วงอายุที่เพิ่งจบไป

2.2 ถ้าเวลาที่ออบเจกต์เกิดในช่วงอายุล่าสุดมีค่าไม่เท่ากับ 0 ซึ่งหมายความว่าออบเจกต์นี้เคยมีการเปลี่ยนแปลงเวลาเกิดมาแล้ว ให้ทำการกำหนดเวลาในการเกิดครั้งใหม่นี้โดยหาจากค่าเฉลี่ยระหว่างเวลาปัจจุบันที่มีการร้องขอนี้กับเวลาในการร้องขอครั้งล่าสุดก่อนหน้าการร้องขอปัจจุบัน. เมื่อได้เวลาในการเกิดครั้งใหม่นี้แล้วก็สามารถนำไปคำนวณอายุของออบเจกต์ในช่วงนี้ได้จากการนำค่าเวลาในการเกิดที่เพิ่งจะคำนวณได้ในครั้งใหม่นี้ลบด้วยเวลาในการเกิดครั้งล่าสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นให้ทำการกำหนดค่า การร้องขอครั้งล่าสุดของ ออบเจกต์นี้เท่ากับเวลาปัจจุบันและทำการบันทึกข้อมูล URL นี้และปิด Record ปัจจุบัน (จบช่วงอายุ) จากนั้นให้ทำการเปิด Record ใหม่ในฐานข้อมูลในฐานะของออบเจกต์เกิดใหม่ (ช่วงอายุใหม่) โดยกำหนดให้ค่า เวลาในการเกิดของออบเจกต์ให้มีค่าเท่ากับที่คำนวณได้จากข้างต้น และกำหนดให้ค่าการร้องขอครั้งล่าสุดของออบเจกต์ในอายุใหม่นี้มีค่าเท่ากับเวลาปัจจุบันของการร้องขอออบเจกต์นี้ในช่วงอายุที่เพิ่งจบไป

3. Life Unknown (LU) ให้นำ URL ที่ถูกร้องขอนั้นไปหาในฐานข้อมูลว่ามี URL นั้นอยู่ในฐานข้อมูลหรือไม่ ถ้าไม่พบ แสดงว่าเป็น URL ใหม่ที่ไม่เคยมีการเก็บข้อมูลมาก่อน ให้ทำการเก็บบันทึกข้อมูล URL, เวลา ลงในฐานข้อมูลและให้กำหนดระดับความแม่นยำเท่ากับ 2 (Accuracy Level =2) ถ้าพบว่ามี URL อยู่แล้วให้ทำการเปรียบเทียบว่าขนาดของ URL ปัจจุบันมีค่าเท่ากับขนาดของ URL ที่พบในฐานข้อมูลหรือไม่ ถ้าขนาดเท่ากัน ให้ไปอ่านข้อมูลบรรทัดต่อไปจาก Access.log และเริ่มต้นการทำงานที่จัดประเภทต่อไป ถ้าขนาดไม่เท่ากัน (มีการเปลี่ยนช่วงอายุเกิดขึ้น) ให้กำหนดระดับความแม่นยำเท่ากับ 2 (Accuracy Level =2) แล้วไปทำงานต่อในกรณีของ Life Miss Case (LM)

3.3 การวิเคราะห์ความผิดพลาดที่เกิดจากอัลกอริธึมในการวัดอายุและขนาดของออบเจกต์

การหาค่าความผิดพลาดที่เกิดจากอัลกอริธึมในการวัดอายุของออบเจกต์

ในงานวิจัยฉบับนี้จะมุ่งเสนอวิธีการหาความผิดพลาดในการคำนวณหาอายุจากการร้องขอที่มีความแม่นยำในระดับที่ 1 (Accuracy Level =1) เท่านั้น

จากรูปที่ 3.1 ข้างต้นเราสามารถคำนวณหาอายุของออบเจกต์ได้ตามสมการ (3.1) ซึ่งเป็นการคำนวณหาอายุจากค่าเฉลี่ย โดยที่เราไม่สามารถที่จะวัดค่าอายุจริง (Lifeactual) ของออบเจกต์นั้นได้โดยทราบแต่เพียงว่าจะมีค่าอยู่ระหว่างค่า (LH_i - LM_{i-1}) และ (LM_i - LH_{i-1}) นั่นคือ

$$\left(LH_i - LM_{i-1}\right) \leq Lifeactual_i \leq \left(LM_i - LH_{i-1}\right) \tag{3.2}$$

จากสมการ (3.1) จะหาค่าอายุของออบเจกต์ในช่วงอายุที่ i จะได้ว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Lifecal_i = \frac{1}{2}\left[\left(LM_i - LH_{i-1}\right) + \left(LH_i - LM_{i-1}\right)\right] \tag{3.3}$$

ดังนั้นเราสามารถหาค่าความผิดพลาดที่เกิดจากการหาอายุของออบเจกต์ที่ได้จากการคำนวณ (ΔL) จากการนำค่าอายุที่คำนวณได้ไปลบออกจากอายุจริงของออบเจกต์ นั่นคือ

$$\Delta L = Lifeactual_i - Lifecal_i \qquad (3.4)$$

นำสมการ (3.2) และ (3.3) มาแทนค่าในสมการ (3.4) จะได้

$$\left(\frac{(LH_i - LM_{i-1})}{2} - \frac{(LM_i - LH_{i-1})}{2}\right) \le \Delta L \le \left(\frac{(LM_i - LH_{i-1})}{2} - \frac{(LH_i - LM_{i-1})}{2}\right) \qquad (3.5)$$

พิจารณานิพจน์ทางซ้ายของอสมการ (3.5) จะได้เท่ากับ

$$= -\frac{1}{2}\left[(LM_{i-1} - LH_{i-1}) + (LM_i - LH_i)\right] \qquad (3.5.1)$$

พิจารณานิพจน์ทางขวาของอสมการ (3.5) จะได้

$$= \frac{1}{2}\left[(LM_{i-1} - LH_{i-1}) + (LM_i - LH_i)\right] \qquad (3.5.2)$$

ดังนั้นจะได้ว่า

$$-a_i \le \Delta L \le a_i \qquad (3.6)$$

เมื่อ

$$a_i = \frac{1}{2}\left[(LM_{i-1} - LH_{i-1}) + (LM_i - LH_i)\right] \qquad (3.7)$$

จากสมการ (3.4) และ (3.6) จะได้ว่า

$$Lifecal_i = Lifeactual_i \pm a_i \qquad (3.8)$$

นั่นก็คือค่าอายุช่วงที่ i ของออบเจกต์ที่คำนวณได้จะมีค่าเท่ากับอายุจริงของออบเจกต์ในช่วงที่ i บวกกับค่าความแม่นยำที่ช่วงอายุที่ i (a_i)

การวิเคราะห์ผลกระทบของอัตราการร้องขอข้อมูล ที่มีผลต่อค่าความผิดพลาดของอายุที่ได้จากอัลกอริธึมในการวัดอายุของออบเจกต์

พิจารณาค่า a จากสมการ (3.7) จะพบว่าค่า a คือค่าเฉลี่ยของ ($LM_{i-1} - LH_{i-1}$) และ ($LM_i - LH_i$) ซึ่งแต่ละพจน์ก็คือช่วงห่างของแต่ละการร้องขอที่มีเข้ามาในขณะที่มีการเปลี่ยนช่วงอายุของออบเจกต์ ถ้าช่วงห่างของการร้องขอมากจะทำให้โอกาสที่จะพบการ

เปลี่ยนช่วงอายุของออบเจกต์มีน้อยลงซึ่งผลที่ตามมาคือค่าอายุที่วัดได้จะมีโอกาสผิดพลาดมากขึ้น ดังนั้นเราสามารถหาค่า a_i ในเชิงสถิติได้จากอัตราส่วนของจำนวนการร้องขอที่มีเข้ามาในแต่ละช่วงอายุต่ออายุจริงของออบเจกต์จาก

$$a_i = \frac{Lifeactual_i}{N+1} \tag{3.9}$$

โดยที่

N คือจำนวนของการร้องขอประเภท LH ที่มีเข้ามาในช่วงระหว่าง LM_{i-1} ถึง LM_i และ ค่า N_i ต้องมีค่ามากกว่า 0 ($N_i > 0$) ทั้งนี้เนื่องจากรูปที่ 1 จะเห็นว่าถ้าค่า N_i เป็น 0 ก็จะทำให้ LH_i หายไปทำให้ไม่สามารถคำนวณหาค่าอายุของออบเจกต์ได้อย่างถูกต้องแม่นยำ

แทนค่า a_i จากสมการ (3.9) ลงในสมการ (3.8) จะได้ว่า

$$Lifecal_i = Lifeactual_i\left(1 \pm \frac{1}{N_i+1}\right) \tag{3.10}$$

จากสมการ (3.9) จะเห็นว่าถ้าออบเจกต์นั้นมีการถูกร้องขอ (N_i) มากขึ้นก็จะทำให้ a_i มีค่าน้อยลงจนเข้าใกล้ 0 และจะทำให้อายุที่คำนวณได้มีค่าแม่นยำมากขึ้นตามสมการ (3.10) นั่นคือ

$$\lim_{N_i \to \infty} Lifecal_i = Lifeactual_i \tag{3.11}$$

จากวิธีการคำนวณหาอายุของออบเจกต์ด้วยวิธีดังกล่าวนั้นเราสามารถที่จะหาความแม่นยำของการคำนวณอายุแต่ละออบเจกต์ออกมาเป็นค่าเปอร์เซ็นต์ของความแม่นยำ (%Accuracy) ได้โดยการกำหนดค่าอายุที่เราทราบแน่นอนและทดลองสร้างการร้องขอออบเจกต์นั้นเพื่อที่จะหาอัตราส่วนของค่า a_i ต่อค่าอายุจริง (ที่เรากำหนดขึ้น)

$$\%acc = \pm\left(\frac{a_i}{Lifeactual_i}\right) x100\% \tag{3.12}$$

แทนค่า a_i จากสมการ (3.9) จะได้ว่า

$$\%acc = \pm\left(\frac{1}{N_i+1}\right) x100\% \tag{3.13}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การทดลองวัดค่าอายุและขนาดของออบเจกต์

จากอัลกอริธึมในการวัดค่าอายุของออกเจกต์ข้างต้นจึงมาทำการทดลองเพื่อทดลองเพื่อทดสอบและหาค่าความผิดพลาดของอายุที่วัดได้จากอัลกอริธึมเทียบกับอายุจริงที่กำหนด โดยจะใช้ออบเจกต์คงที่ออบเจกต์หนึ่งบนเว็บเซิร์ฟเวอร์เป็นออบเจกต์ที่จะใช้วัดอายุ จากนั้นจะการสร้างการร้องขอออบเจกต์ดังกล่าวจากไคลเอนท์ผ่านไปยังพร็อกซีเซิร์ฟเวอร์ ด้วยอัตราการร้องขอต่าง ๆ และทำการการวัดค่าอายุของออบเจกต์นั้นผ่านทางล็อกไฟล์ของพร็อกซีเซิร์ฟเวอร์ซึ่งใช้ Squid ในการทำงาน โดยในขณะที่ทำการสร้างการร้องขอจากไคลเอนท์ไปยังพร็อกซีเซิร์ฟเวอร์นั้นจะมีการควบคุมให้ออบเจกต์ที่ถูกร้องขอนั้นมีการเปลี่ยนแปลงอายุในช่วงเวลาที่คงที่เป็นจำนวน 3 ช่วงอายุ

การทดลองที่ 3.1 เป็นการทดลองสร้างการร้องขอที่มีอัตราคงที่ต่อออบเจกต์ที่มีอายุคงที่เป็นจำนวน 3 ช่วงอายุ โดยแปรค่าอายุตั้งแต่ 30, 60, 90, 120, 150, 180, 210, 240, 270 และ 300 วินาที (เนื่องจากต้องการประหยัดเวลาในการทดลองจึงเลือกกำหนดช่วงอายุของออบเจกต์ที่มีค่าน้อย ๆ) จำนวน 10 ค่า และแปรจำนวนการร้องขอ 0.8, 0.9, 1, 1.18, 1.33, 2, 4, 10, 20 และ 100 ครั้งต่อ 1 ช่วงอายุรวมเป็นจำนวน 10 ค่า โดยนำค่าอายุที่คำนวณได้ 3 ค่าจากในแต่ละช่วงอายุมาทำการเฉลี่ยก่อนนำไปหาความผิดพลาดจากช่วงอายุจริง และทำการทดลองข้างต้นซ้ำเหมือนเดิมอีก 1 ชุด แต่จะสร้างการร้องขอแบบสุ่มที่มีค่าเฉลี่ยของอัตราการร้องขอตั้งแต่ 0.8, 0.9, 1, 1.18, 1.33, 2, 4, 10, 20 และ 100 ครั้งต่อ 1 ช่วงอายุรวมเป็นจำนวน 10 ค่า และนำค่าอายุที่วัดได้จากทั้งสามช่วงอายุมาเฉลี่ยเป็นค่าอายุที่คำนวณได้ (Lifecal) และนำมาทำการนอร์มอลไลซ์เพื่อหาความผิดพลาดจากค่าอายุจริงที่ถูกนอร์มอลไลซ์แล้ว (Normalized Error, NE) ตามสมการ

$$NE_i = \left(\frac{Lifecal_i - Lifeactual_i}{Lifeactual_i} \right) \tag{3.14}$$

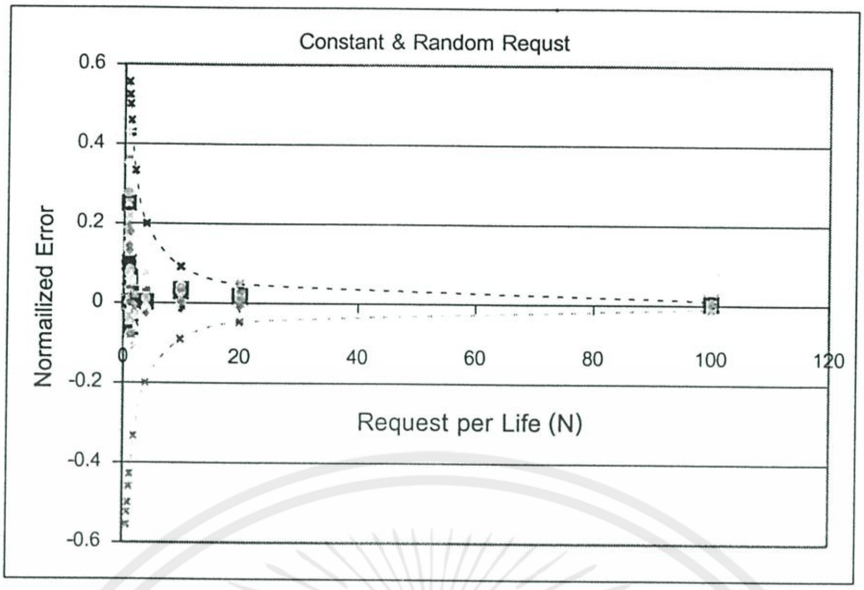
โดยที่

NE_i คือ ค่าความผิดพลาดที่นอร์มอลไลซ์แล้ว (Normalized Error)

$Lifecal_i$ คือ ค่าอายุที่ได้จากการคำนวณตามอัลกอริธึม

$Lifeactual_i$ คือ ค่าอายุจริงที่ได้กำหนดไว้ตอนเริ่มต้นที่จะทำการทดลองวัดค่าอายุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

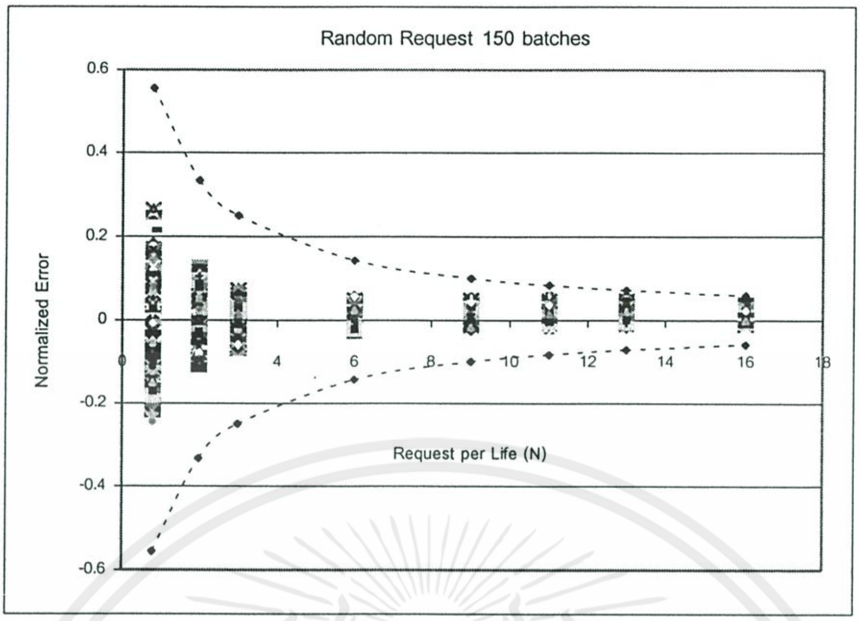


รูปที่ 3.2 ค่าความผิดพลาดของอายุของออบเจกต์ซึ่งถูกนอร์มอลไลซ์ (Normalized) ณ.อัตราการร้องขอต่าง ๆ

จากรูปที่ 3.2 จะเห็นได้ว่าเมื่อจำนวนครั้งในการร้องขอต่อ 1 ช่วงอายุของออบเจกต์เพิ่มขึ้น ค่าความผิดพลาดที่เกิดขึ้นก็จะมีช่วงของความผิดพลาดเข้าใกล้ 0 มากขึ้นหรือมีความผิดพลาดน้อยลง นั้นหมายถึงว่ายิ่งมีจำนวนการร้องขอต่อ 1 ช่วงอายุของออบเจกต์มากขึ้นค่าอายุที่คำนวณที่ได้ก็จะมีความแม่นยำมากขึ้น และจะสังเกตค่าความผิดพลาดที่ปรากฏจากกราฟว่ามีค่าอยู่ในกรอบของค่า a จากสมการ (3.13) ซึ่งแสดงด้วยเส้นประในรูปกราฟ

การทดลองที่ 3.2 ทำการทดลองหาค่าความผิดพลาดในการวัดอายุของออบเจกต์จากการคำนวณ โดยทำการทดลองซ้ำในแต่ละช่วงอายุตั้งแต่ 5, 10, 20, 50, 100, 120 และ 150 วินาที เป็นจำนวนช่วงอายุละ 150 ครั้ง โดยมีการแปรอัตราการร้องขอแบบสุ่มที่มีค่าเฉลี่ยอยู่ที่ 0.8, 2, 3, 6, 9, 11, 13 และ 16 ครั้งต่อ 1 ช่วงอายุแล้วนำความผิดพลาดของอายุที่คำนวณได้มาทำการนอร์มอลไลซ์ตามสมการที่ (3.14)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

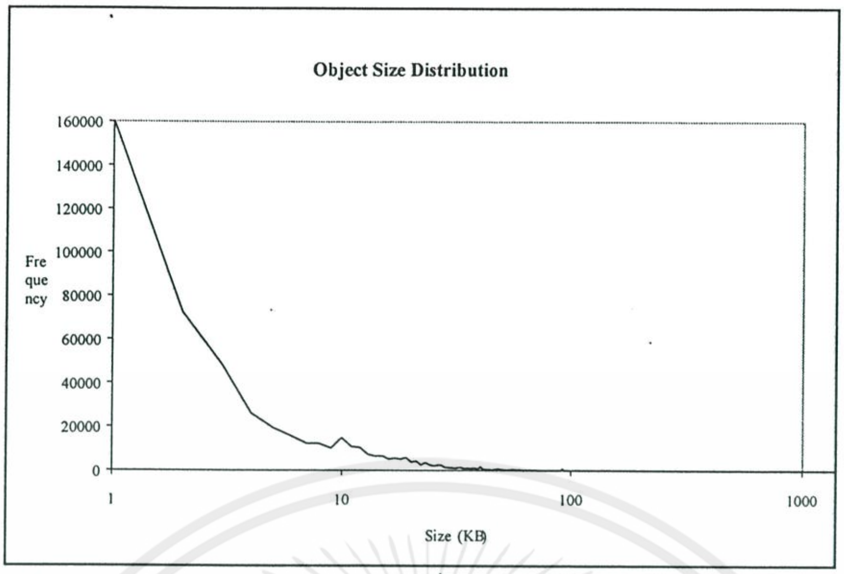


รูปที่ 3.3 ค่าความผิดพลาดของอายุที่คำนวณซึ่งถูกนอร์มอลไลซ์ ณ.อัตราการร้องขอใน 1 ช่วงอายุต่าง ๆ

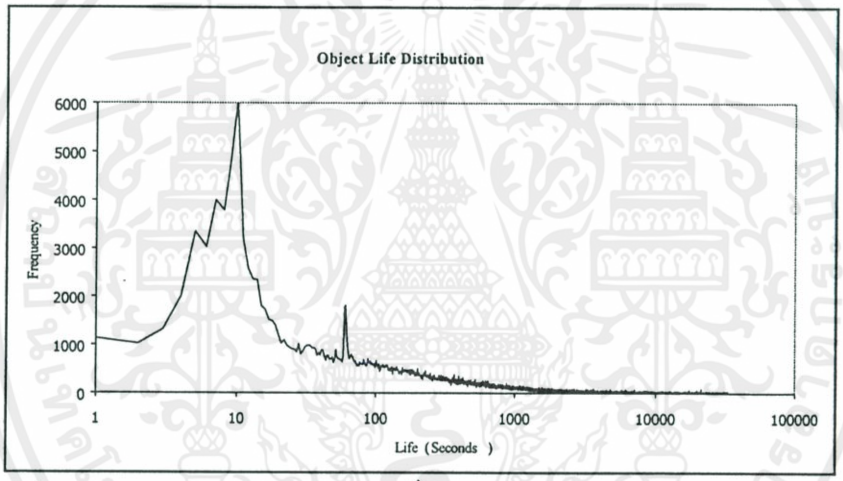
จากผลการทดลองที่ 3.2 ซึ่งเป็นการทดลองเพื่อที่มีจุดประสงค์เพื่อทำการวัดค่าอายุของออบเจกต์เป็นจำนวนมากเพื่อแสดงขอบเขตของความผิดพลาดในการวัดค่าอายุดังแสดงในกราฟรูปที่ 3.3 จะเห็นได้ว่าอายุของออบเจกต์ที่วัดได้ยังมีค่าความผิดพลาดอยู่ในช่วงที่ระบุไว้คือไม่เกินค่า a_i จากสมการ (3.13) ซึ่งแสดงด้วยเส้นประในรูปและ ถ้าอัตราการร้องขอมีความถี่มากขึ้นจะทำให้ค่าความผิดพลาดของค่าอายุที่วัดได้ลดลง

การทดลองที่ 3.3 จากวิธีการหาอายุและขนาดของออบเจกต์ข้างต้นได้นำมาพัฒนาโปรแกรมเพื่อทดลองวัดอายุของออบเจกต์จริงที่ถูกร้องขอผ่านเครื่องพร็อกซีเซิร์ฟเวอร์ โดยใช้ PERL Script [21] เป็นเครื่องมือเพื่อช่วยในการพัฒนาโปรแกรมเพื่อประมวลผล ซึ่งในการทดลองนี้จะใช้ PERL Script ที่พัฒนาขึ้นมาประมวลผลล็อกไฟล์จากเครื่อง proxy1.kmitl.ac.th ซึ่งทำหน้าที่เป็นพร็อกซีเซิร์ฟเวอร์หลักของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง ในช่วงเวลาตั้งแต่วันที่ 1 ถึง 14 กุมภาพันธ์ พ.ศ.2543 ซึ่งคิดเป็นข้อมูลการร้องขอ 9,422,162 requests (เฉลี่ยวันละ 673,011 requests) มาทำการหาการกระจายของขนาดและอายุของออบเจกต์ ได้ผลดังแสดงในรูปที่ 3.4 และ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 กราฟการกระจายของขนาดของออบเจกต์



รูปที่ 3.5 กราฟการกระจายของอายุของออบเจกต์

ผลการทดลองแสดงให้เห็นถึงการกระจายตัวของขนาดของออบเจกต์จำนวนร้อยละ 86.82 จะมีค่าไม่เกิน 10 กิโลไบต์ โดยมีค่าเฉลี่ยอยู่ที่ 8.78 กิโลไบต์ ในส่วนการกระจายอายุของออบเจกต์ส่วนมากมีค่าน้อยกว่า 10 วินาทีโดยมีค่าเฉลี่ยอยู่ที่ 19128.29 วินาที (ประมาณ 5 ชั่วโมง 19 นาที) โดยมีความผิดพลาดเฉลี่ยทุกออบเจกต์ที่มี Accuracy Level 1 อยู่ที่ ±26.69% ทั้งนี้มี ออบเจกต์จำนวน 347,440 ออบเจกต์ที่มีขนาดอยู่ระหว่าง 0 ถึง 1 กิโลไบต์ และอีก 885 ออบเจกต์ที่มีอายุอยู่ระหว่าง 0 ถึง 1 วินาที เนื่องจากในความเป็นจริงมีออบเจกต์บางตัวที่มีขนาดไม่ถึง 1 กิโลไบต์ (เช่น มีขนาด 250 ไบต์) และ/หรือ มีอายุไม่ถึง 1 วินาที

อนึ่งการคำประมวลผลจากล็อกไฟล์ในการทดลองที่ 3-3 ดังกล่าวทำงานบนเครื่อง HP LH4 Dual Pentium III Xeon 500MHz ใช้เวลาในการประมวลผลรวม 172 ชั่วโมง

3.5 สรุป

เราสามารถวัดอายุของออบเจกต์ได้จากล็อกไฟล์ของ Squid ตามอัลกอริธึมในหัวข้อวิธีการหาอายุและขนาดของออบเจกต์และระบุความแม่นยำของการวัดได้ตามสมการ (3.13) โดยที่ค่าความผิดพลาดที่ได้จากการวัดอายุของออบเจกต์จะมีค่าลดลงเมื่อมีอัตราการร้องขอออบเจกต์ในช่วงอายุนั้น ๆ มากขึ้น นอกจากนี้แล้วยังสามารถนำวิธีการวัดค่าอายุของออบเจกต์นี้ไปใช้กับการวัดค่าจากล็อกไฟล์ประเภทอื่น ๆ ที่มีเวลากำกับและสามารถจัดประเภทของแต่ละบรรทัดของล็อกไฟล์ได้ว่ามีความเกี่ยวข้องกับค่าที่ต้องการจะวัดหรือไม่ รวมทั้งสามารถหาค่าความผิดพลาดของค่าที่วัดได้อีกด้วย โดยเราสามารถนำค่าอายุของออบเจกต์ที่วัดได้นี้ไปทำการปรับตั้งค่าอายุอ้างอิงได้ดังจะได้กล่าวถึงในบทถัดไป

ในส่วนของการทำการทดลองวัดค่าขนาดและอายุจริงของออบเจกต์ในอินเทอร์เน็ตที่ถูกร้องขอผ่านเครื่องพร็อกซีเซิร์ฟเวอร์นั้นปรากฏว่าออบเจกต์ส่วนใหญ่มีขนาดเล็กและมีการเปลี่ยนแปลงอายุที่บ่อยมาก ดังจะเห็นได้จากออบเจกต์ส่วนใหญ่มีขนาดต่ำกว่า 10 กิโลไบต์ และออบเจกต์ส่วนใหญ่ก็มีช่วงอายุต่ำกว่า 100 วินาที ซึ่งเป็นดัชนีสะท้อนพฤติกรรมของเนื้อหาที่มีอยู่ในอินเทอร์เน็ตว่ามีขนาดเล็กและมีการเปลี่ยนแปลงที่รวดเร็วมากจนอาจจะทำให้ความต้องการพื้นที่ในการทำดิสก์แคชนั้นไม่ต้องใหญ่มากอย่างในปัจจุบันเพราะเนื่องจากขนาดของออบเจกต์ส่วนใหญ่มีขนาดเล็กและมีการเปลี่ยนแปลงอยู่ตลอดเวลาจึงทำให้จะมีการนำออบเจกต์เข้าและออกจากดิสก์แคชอยู่ตลอดเวลา ซึ่งเราสามารถนำค่าขนาดเฉลี่ยของออบเจกต์นี้ไปใช้ในการคำนวณเพื่อปรับตั้งค่า cache_dir ได้ดังที่ได้กล่าวมาแล้วในบทที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การปรับแต่งพร็อกซีเซิร์ฟเวอร์

ในบทนี้จะเสนอการทดลองเพื่อปรับแต่งค่าพารามิเตอร์ของพร็อกซีเซิร์ฟเวอร์ 2 ส่วนคือในส่วนแรกจะเป็นการทดลองหาค่าขนาดของดิสก์แคชที่จะใช้ในการปรับตั้งให้กับพร็อกซีเซิร์ฟเวอร์ตามพฤติกรรมการใช้งานของผู้ใช้ที่เปลี่ยนไป ณ.ช่วงเวลาหนึ่ง ๆ โดยจะนำเสนออัลกอริธึมในการคำนวณหาค่าขนาดของดิสก์แคชที่เหมาะสมจากการประมาณค่าจากความสัมพันธ์ระหว่างค่าอัตราการพบข้อมูลในแคช (H) ซึ่งมีคิดเป็นเปอร์เซ็นต์ของการร้องขอที่พบข้อมูลต่อจำนวนการร้องขอทั้งหมด และค่าขนาดของดิสก์แคชที่ใช้ในช่วงเวลาที่ผ่านมา ในส่วนของการทดลองที่สองจะเป็นการทดลองปรับค่า reference_age ของพร็อกซีเซิร์ฟเวอร์เพื่อศึกษาผลกระทบต่อขนาดของดิสก์แคชและอัตราการพบข้อมูลของแคช

4.1 พารามิเตอร์ที่เกี่ยวข้องกับการปรับแต่งพร็อกซีเซิร์ฟเวอร์

ส่วนประกอบที่สำคัญของเครื่องคอมพิวเตอร์ที่ทำหน้าที่พร็อกซีเซิร์ฟเวอร์ส่วนหนึ่งคืออุปกรณ์ดิสก์ที่ใช้เก็บข้อมูลซึ่งเป็นส่วนประกอบที่มีราคาสูงโดยเฉพาะเมื่อระบบดิสก์นั้นมีความจุข้อมูลสูงและประกอบไปด้วยฮาร์ดดิสก์จำนวนมาก ดังนั้นพารามิเตอร์ขนาดของดิสก์แคชจึงเป็นพารามิเตอร์หนึ่งที่น่าสนใจในการจะหาค่าที่เหมาะสม เนื่องจากว่าถ้าพร็อกซีเซิร์ฟเวอร์มีพื้นที่ในการจัดเก็บออบเจกต์ในแคชน้อยก็จะทำให้โอกาสในการพบออบเจกต์ที่ถูกร้องขอในแคชนั้นลดลงไปเป็นผลให้พร็อกซีเซิร์ฟเวอร์ต้องไปร้องขอออบเจกต์จากเซิร์ฟเวอร์ต้นทาง ซึ่งจะทำให้เวลาในการตอบสนอง (Response time) เพิ่มมากขึ้น ในทางกลับกันการเพิ่มพื้นที่ที่ใช้ในการเก็บออบเจกต์ก็จะทำให้มีพื้นที่สำหรับจัดเก็บจำนวนออบเจกต์ได้มากขึ้นซึ่งจะทำให้มีโอกาสที่จะพบออบเจกต์ที่ถูกร้องขออยู่ในแคชมากขึ้น (ค่า H เพิ่มขึ้น) แต่การเพิ่มขนาดของดิสก์แคชนั้นจะต้องเสียค่าใช้จ่ายในการจัดหาระบบดิสก์ที่มีขนาดใหญ่ขึ้น และเมื่อทำการเพิ่มขนาดของพื้นที่ของดิสก์ (S) ขึ้นเรื่อย ๆ จะพบว่าอัตราการเพิ่มขึ้นของอัตราการพบข้อมูลในแคชจะเริ่มมีค่าลดลง และมีค่าเข้าใกล้ค่าคงที่ค่าหนึ่ง [15]

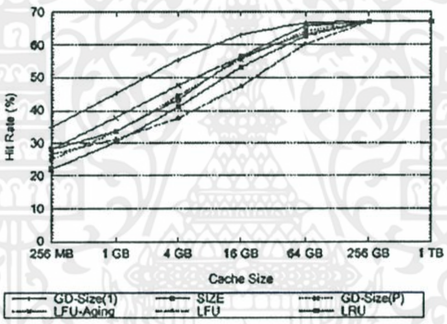
พารามิเตอร์อีกตัวหนึ่งที่จะทำการปรับคือค่าอายุอ้างอิง (reference_age) ซึ่งจะเป็นค่าเวลาที่มากที่สุดที่พร็อกซีเซิร์ฟเวอร์จะเก็บออบเจกต์หนึ่ง ๆ ไว้ในแคชของมัน ซึ่งถ้ามีค่าน้อยเกินไปก็จะทำให้มีการลบออบเจกต์ออกจากแคชบ่อยขึ้น โอกาสที่จะพบข้อมูลในแคชก็จะน้อยลง แต่ถ้ามีค่ามากเกินไปก็จะทำให้ต้องการพื้นที่ในการจัดเก็บมากขึ้น โดยในการทดลองปรับค่านี้จะมีการนำค่าอายุเฉลี่ยของ ออบเจกต์ในอินเทอร์เนตที่ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์ที่ได้จากทดลองวัดค่าอายุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และขนาดของออบเจกต์โดยใช้ล็อกไฟล์จากเครื่องพร็อกซีเซิร์ฟเวอร์ที่ใช้งานจริงมาทำลองปรับค่าด้วย

4.2 อัลกอริธึมในการปรับแต่งค่าดิสก์แคช

การปรับแต่งพร็อกซีเซิร์ฟเวอร์เพื่อหาขนาดของดิสก์แคชที่เหมาะสมนั้นจะอาศัยเกณฑ์การวัดจากการปรับค่าขนาดของดิสก์แคช (s) ที่ทำให้มีอัตราการพบข้อมูลในแคช (H) ซึ่งมีค่าเป็นเปอร์เซ็นต์ของการร้องขอที่พบข้อมูลต่ออัตราการร้องขอทั้งหมด โดยจะพยายามหา s ที่คุ้มค่าโดยจะวัดความคุ้มค่าในรูปของอัตราการเพิ่มของค่า H ที่ได้จากการเพิ่มค่า s ซึ่งโดยธรรมชาติค่าของ H จะมีค่าเท่ากับ 0 เมื่อค่า s เท่ากับ 0 นั่นคือไม่มีการทำแคช และเมื่อทำการเพิ่มขนาดของแคชค่า H ก็จะเพิ่มขึ้นและเมื่อทำการเพิ่มค่า s ไปเรื่อย ๆ ค่า H ก็จะเริ่มลู่เข้าหาค่าคงที่หนึ่ง [9] ดังแสดงในรูปที่ 4.1



รูปที่ 4.1 ความสัมพันธ์ของอัตราการพบข้อมูลและขนาดของดิสก์แคช [9]

เพื่อที่จะวัดความคุ้มค่าในการใช้พื้นที่ของดิสก์แคชจึงได้กำหนดค่า R ขึ้นมาเพื่อใช้เป็นดัชนีในการวัดความคุ้มค่าต่อไป โดยให้ค่า R คืออัตราส่วนระหว่าง H ต่อ S ($R=\frac{H}{S}$) และมีสมมุติฐานเริ่มแรกในการปรับค่าขนาดของดิสก์ให้ได้ค่า R ที่มีค่ามากที่สุดเท่าที่จะเป็นไปได้ หรือพยายามหาค่า S ที่ทำให้ค่า H มีค่าสูงมากสุดที่จะยังคงทำให้ค่า R มากที่สุด หรืออาจกล่าวได้ว่าอัตราการเปลี่ยนแปลงของค่า H เมื่อเทียบกับ S มีค่าเท่ากับอัตราการเปลี่ยนแปลงของค่า R เมื่อเทียบกับ S (ขนาดเท่ากันแต่เครื่องหมายตรงกันข้าม) นั่นคือ

$$\frac{\partial H}{\partial s}+s\frac{\partial\left(\frac{H}{s}\right)}{\partial s}=0 \tag{4.1}$$

จากสมการ (4.1) ทำการหาค่า $\frac{\partial\left(\frac{H}{s}\right)}{\partial s}$ จะได้ว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{\partial\left(\frac{H}{s}\right)}{\partial s}=\frac{1}{s^2}\left[s\frac{\partial H}{\partial s}-H\frac{\partial s}{\partial s}\right]$$

$$\frac{\partial\left(\frac{H}{s}\right)}{\partial s}=\frac{1}{s}\frac{\partial H}{\partial s}-\frac{H}{s^2}$$ โดยที่ $s\neq 0$ (4.2)

แทนค่าสมการ (4.2) ใน สมการ (4.1) จะได้

$$\frac{\partial H}{\partial s}+s\cdot\left[\frac{1}{s}\cdot\frac{\partial H}{\partial s}-\frac{H}{s^2}\right]=0$$

$$\frac{\partial H}{\partial s}+\frac{\partial H}{\partial s}-\frac{H}{s}=0$$

$$2\cdot\frac{\partial H}{\partial s}-\frac{H}{s}=0$$

$\frac{\partial H}{\partial s}=\frac{1}{2}\cdot\frac{H}{s}$ นั่นคือ S ที่ทำให้ $\frac{\partial H}{\partial s}=\frac{R}{2}$ โดยที่ $R=\frac{H}{s}$

จากสมการ (4.1) ถ้ากำหนดให้ $g(s)=\frac{\partial H}{\partial s}+s\cdot\frac{\partial\left(\frac{H}{s}\right)}{\partial s}=0$ (4.3)

หรือเขียนได้ว่า $g(s)=0$ ซึ่งจากการวิเคราะห์สมการ (4.3) พบว่าจะเกิดบางกรณีที่ไม่สามารถหาค่า S ที่ทำให้ $g(s)=0$ ได้

แต่เนื่องจากค่า $g(s)$ เป็นอัตราการเปลี่ยนแปลงของ H และ $\frac{H}{s}$ เทียบกับค่า s ดังนั้นจุดที่เหมาะสมจึงควรจะเป็นค่า s ที่ทำให้ $\frac{\partial g(s)}{\partial s}=0$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากค่า $g(s)$ ที่กำหนดในสมการ (4.3) จะได้ว่า

$$g(s)=2\cdot\frac{\partial H}{\partial s}-\frac{H}{s}$$

ดังนั้น

$$\frac{\partial g(S)}{\partial s}=\frac{\partial}{\partial s}\left[2\cdot\frac{\partial H}{\partial s}-\frac{H}{s}\right]$$

$$\frac{\partial g(S)}{\partial s}=2\cdot\frac{\partial^2 H}{\partial s^2}-\frac{\partial\left(\frac{H}{s}\right)}{\partial s}$$

แทนค่า $\frac{\partial\left(\frac{H}{s}\right)}{\partial s}$ จากสมการ (4.2)

$$\frac{\partial g(S)}{\partial s}=2\cdot\frac{\partial^2 H}{\partial s^2}-\left[\frac{1}{s}\cdot\frac{\partial H}{\partial s}-\frac{H}{s^2}\right]$$

$$\frac{\partial g(S)}{\partial s}=2\cdot\frac{\partial^2 H}{\partial s^2}-\frac{1}{s}\cdot\frac{\partial H}{\partial s}+\frac{H}{s^2} \tag{4.4}$$

ในทางปฏิบัติจริงจะไม่สามารถทราบค่าที่แน่นอนของ H ซึ่งเป็นฟังก์ชั่นที่จะอยู่ในรูปของค่า S และพารามิเตอร์อื่น ๆ เช่น พฤติกรรมการใช้งานของผู้ใช้ที่เปลี่ยนไปตามเวลา ($H=f(S,t)$) แต่จากงานวิจัย [9] ทำให้ทราบถึงลักษณะของ H เมื่อเทียบกับ S ได้และเพื่อเป็นการสะดวกในการคำนวณอีกทั้งเพื่อเป็นต้นแบบเบื้องต้นของวิธีการและอัลกอริธึทในงานวิจัยนี้ จึงจะทำการประมาณค่า H ในรูปของสมการเอ็กซ์โพเน็นเชียลอย่างง่าย (Simple exponential function) โดยให้ค่า H มีค่าสูงสุดเท่ากับ 1 เมื่อค่า S เข้าใกล้อนันต์ ดังแสดงในสมการ

$$H=h(S,t)=1-b_t\cdot e^{-a_t S_t} \tag{4.5}$$

โดยการหาค่า a และ b ในสมการ (4.6) สามารถหาได้จากทำการเก็บข้อมูล H และ S ย้อนหลังไปเป็นจำนวน L คู่ลำดับ จากนั้นค่า H และ S มาทำการหาค่า a และ b ด้วย Exponential Regression ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$a=\left[\frac{L\cdot\sum_{i=1}^{L}x_i\cdot \ln y_i-\sum_{i=1}^{L}x_i\cdot\sum_{i=1}^{L}\ln y_i}{L\cdot\sum_{i=1}^{L}x_i^2-\left(\sum_{i=1}^{L}x_i\right)^2}\right] \text{ และ } b=exp\left(\left[\frac{\sum_{i=1}^{L}\ln y_i-a\cdot\sum_{i=1}^{L}x_i}{L}\right]\right) \quad (4.6)$$

โดยที่ค่า x_i คือ ค่า S ที่เวลาต่าง ๆ

y_i คือ ค่า H ที่เวลาต่าง ๆ

นำค่า a และ b ที่คำนวณได้จากสมการ (4.6) แทนในสมการ (4.5) โดยถือว่าตลอดช่วงเวลาที่นำค่า H และ S มาใช้ในการประมาณค่านั้นพฤติกรรมการใช้งานของผู้ใช้งานนั้นมีค่าคงที่

$$H=h(S)=1-b\cdot e^{-aS} \quad (4.7)$$

แทนค่าสมการ (4.7) ในสมการ (4.4) และให้มีค่าเท่ากับ 0 เพื่อหาจุดวิกฤต (Critical point) ของฟังก์ชันจะได้ว่า

$$f(S)=-2a^2b\cdot e^{-aS}-\frac{1}{S}\cdot ab\cdot e^{-aS}+\frac{1}{S^2}\cdot\left(1-b\cdot e^{-aS}\right)=0 \quad (4.8)$$

โดยที่กำหนดให้ $f(S)=\frac{\partial g(S)}{\partial S}$

พิจารณาสมการที่ (4.8) เราต้องการหาค่า S ที่ทำให้ $f(S)=0$ แต่เนื่องจาก $f(S)$ อยู่ในรูปของสมการที่ซับซ้อนไม่สามารถหารูปแบบสามัญ (General Form) ได้ และเพื่อเป็นการเตรียมการเผื่อไว้สำหรับอนาคตที่รูปแบบของ $f(S)$ ที่อาจจะซับซ้อนมากขึ้น จึงจะทำการประมาณค่า S โดยใช้ Numerical Analysis เพื่อทำให้อัลกอริธึมนี้สามารถใช้กับสมการแบบทั่วไปได้ โดยเลือกวิธีของ Secant หาค่าของ x ที่ทำให้ $f(x)=0$ โดย

$$X_1=\frac{X_{-1}\cdot f(X_0)-X_0\cdot f(X_{-1})}{f(X_0)-f(X_{-1})}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั่นคือ

$$X_i=\frac{X_{i-1}\cdot f(X_i)-X_i\cdot f(X_{i-1})}{f(X_i)-f(X_{i-1})}$$ เมื่อ i =Iteration

ให้ $f(S)=0$ โดยที่ $f(S)=2\cdot\frac{\partial^2 H}{\partial s^2}-\frac{1}{s}\cdot\frac{\partial H}{\partial s}+\frac{H}{s^2}$ (4.9)

กำหนดให้ $H\approx 1-b\cdot e^{-aS}$ เพื่อแทนค่า H ในสมการ (4.9) เนื่องจากเราไม่สามารถทราบค่าที่แท้จริงของ H ได้จึงต้องทำการประมาณค่า H จากสมการ (4.8) จะได้ว่า

$$f(S)=2\cdot\frac{\partial^2 h(S,t)}{\partial s^2}-\frac{1}{s}\cdot\frac{\partial h(s,t)}{\partial s}+\frac{h(s,t)}{s^2} \tag{4.10}$$

ทำการหาค่า a_{ti} และ b_{ti} จากการประมาณค่า (S_t, H_t) ในช่วงเวลาที่ $t=t_i; 0\le i\le L$ ดังนั้นจากสมการ (4.10) จะได้ว่า

$$f(S)=-2a^2\cdot b\cdot e^{-aS}-\frac{1}{S}\cdot a\cdot b\cdot e^{-aS}+\frac{1}{S^2}\cdot\left(1-b\cdot e^{-aS}\right) \tag{4.11}$$

เมื่อต้องการหาค่า S ที่ทำให้ $f(S)=0$ ตาม Secant method จากสมการ (4.11) จะได้

$$S_{i+1}=\frac{S_{i-1}\cdot f(S_i)-S_i\cdot f(S_{i-1})}{f(S_i)-f(S_{i-1})} \tag{4.12}$$

โดยที่ค่า S_{i+1} คือค่าของดิสก์แคชที่จะการปรับตั้ง

จากอัลกอริธึมในการคำนวณหาค่าขนาดของดิสก์แคชที่นำไปปรับค่า(S_{i+1}) สามารถเขียนสรุปเป็นขั้นตอนได้ดังนี้

1. ทำการเก็บค่า H และ S จริงที่เวลาต่างๆ ย้อนหลังไป L ค่าโดยแต่ละค่ามีช่วงห่างที่เท่ากัน และค่า L ควรจะมีค่ามากพอที่จะคลอบคลุมพฤติกรรมการใช้งานในแต่ละวันของผู้ใช้

2. จากนั้นนำค่า H และ S ที่ได้มาทำการประมาณค่าเพื่อหาค่า a และ b ตามสมการ (4.5)

3. นำค่า a และ b ใช้ที่คำนวณได้ไปแทนค่าในสมการ (4.11) เพื่อทำการหาค่า $f(S_i)$ ที่เวลาปัจจุบัน

4. จากนั้นจึงทำการคำนวณค่า $f(S_{i-1})$ เพื่อนำมาแทนค่าในสมการ (4.12) เพื่อหาค่า S_{i+1} ที่จะนำไปใช้ในการปรับตั้งค่าดิสก์แคชของ Squid

5. ในกรณีที่ค่า S_{i+1} ที่คำนวณได้มีค่าเกินขอบเขตของขนาดพื้นที่ในการเก็บข้อมูลบนเครื่องพร็อกซีเซิร์ฟเวอร์ก็จะต้องมีการกำหนดให้ค่าขนาดของดิสก์แคชที่จะทำการปรับจริงนั้นอยู่ที่

ค่าสูงสุดหรือต่ำสุดที่จะยอมรับได้ โดยนำค่าที่ได้จากการพิจารณาร่วมกับขอบเขตนี้ไปทำการปรับตั้งค่าพารามิเตอร์ cache_dir ในไฟล์ squid.conf

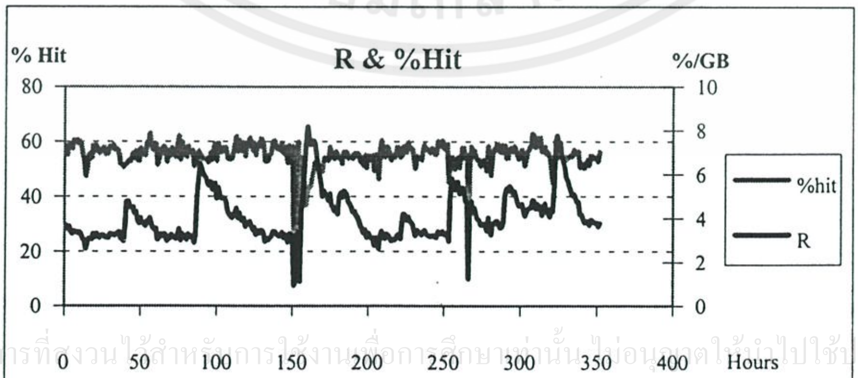
จากอัลกอริธึมนี้ทำให้เราสามารถทำการคำนวณหาค่า S_{i+1} ซึ่งเป็นค่าขนาดของดิสก์แคชที่จะนำไปใช้ในการปรับตั้งให้กับพร็อกซีเซิร์ฟเวอร์ต่อไป

4.3 การทดลองปรับแต่งค่าขนาดของดิสก์แคชและอายุอ้างอิง

4.3.1 การทดลองปรับค่าขนาดของดิสก์แคช

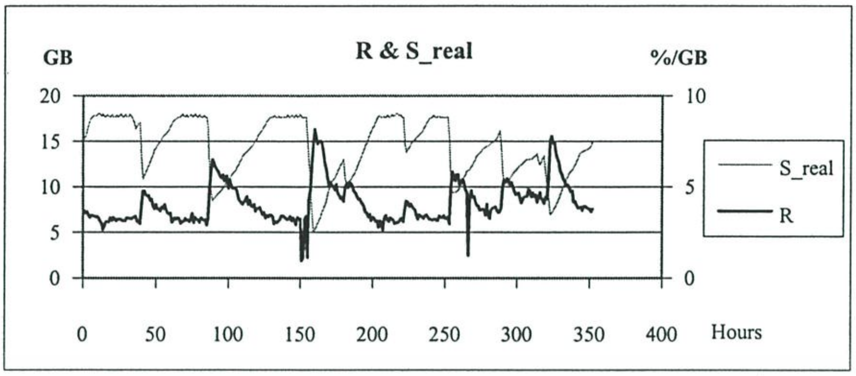
การทดลองนี้เป็นการนำอัลกอริธึมในการหาขนาดของดิสก์แคชที่จะใช้กำหนดให้ Squid มาทดลองกับข้อมูลจริง โดยได้นำอัลกอริธึมดังกล่าวมาพัฒนาด้วย PERL Script เพื่อทำการประมวลผลและทำการทดลองปรับค่าจริงกับเครื่อง proxy.kmitl.ac.th ซึ่งเป็นพร็อกซีเซิร์ฟเวอร์หลักของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง โดยเป็นเครื่อง HP LH4 Plus Dual Pentium-iii Xeon 500 MHz มีหน่วยความจำหลัก 512 เมกะไบต์ ใช้ระบบปฏิบัติการ FreeBSD เวอร์ชั่น 4.1.1 และมีพื้นที่ฮาร์ดดิสก์ขนาด 20 กิกะไบต์สำหรับดิสก์แคชโดยเฉพาะ

ในการคำนวณหาค่า S_{i+1} ซึ่งเป็นค่าขนาดของดิสก์แคชที่เหมาะสมจะการคำนวณทุก ๆ 1 ชั่วโมงโดยทำการเก็บข้อมูลค่าเฉลี่ยของ H และ S ย้อนหลังกลับไป 1 ชั่วโมงเพื่อนำมาใช้ในหาค่า a และ b ตามสมการ $h(S,t)=1-b_{t_i}\cdot e^{-a_{t_i}S_{t_i}}$ ซึ่งจะทำการคำนวณค่า a_{t_i} และ b_{t_i} ใหม่ทุก 1 ชั่วโมงก่อนที่จะนำไปคำนวณหาค่า S_{i+1} โดยค่าที่คำนวณได้จะต้องมีค่าอยู่ระหว่าง 1 ถึง 20 กิกะไบต์ซึ่งเป็นค่าต่ำสุดและสูงสุดที่ยอมให้มีการปรับตั้งได้อันเนื่องมากจากข้อจำกัดของพื้นที่เก็บข้อมูลบนเครื่อง proxy.kmitl.ac.th และทำการทดลองปรับค่า S_{i+1} ทุก 1 ชั่วโมงและเก็บค่า H จริงที่เป็นผลจากการปรับในแต่ละช่วงไว้เพื่อเปรียบเทียบโดยนำมาสร้างเป็นกราฟดังแสดงในรูปที่ 4.2



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

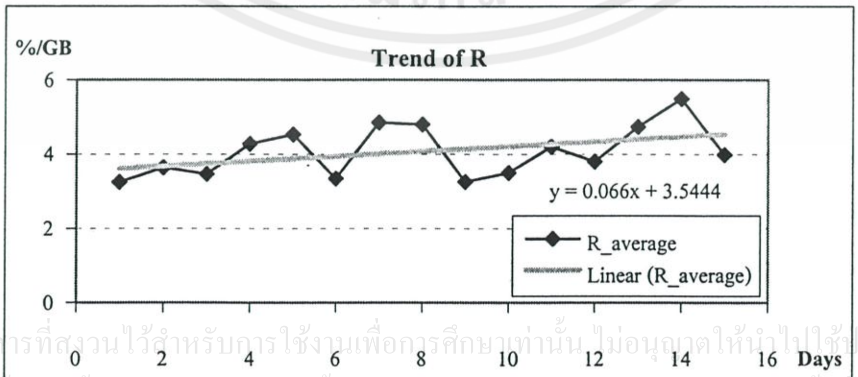
รูปที่ 4.2 ค่า R และ %Hit



รูปที่ 4.3 ค่า R และ S ที่ใช้ในการทำดิสก์แคชจริง (S_real)

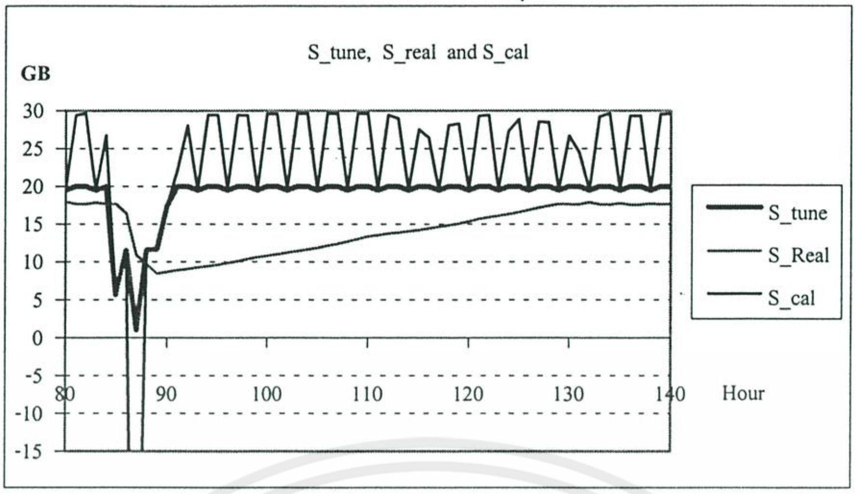
ในรูปที่ 4.2 เป็นการแสดงค่า R ซึ่งก็คืออัตราส่วนของ %Hit ที่วัดได้จากผลการทำงานของพร็อกซีเซิร์ฟเวอร์ต่อค่า S_real ซึ่งเป็นค่าขนาดของดิสก์แคชที่พร็อกซีเซิร์ฟเวอร์ใช้งานจริง โดยแสดงค่าเทียบกับค่า %Hit ที่วัดได้ โดยจากรูปกราฟจะเห็นว่าอัตราการพบข้อมูลในช่วงที่มีการปรับค่า S นั้นค่อนข้างคงที่และมีการเปลี่ยนแปลงขึ้นลงไม่มากนัก ยกเว้นในช่วงประมาณชั่วโมงที่ 150 ของการปรับ จะเห็นว่ามีการลดลงของค่า %Hit อย่างมากและส่งผลให้ค่า R มีค่าเพิ่มขึ้นสูงมาก ทั้งนี้เนื่องจากในช่วงเวลาดังกล่าวเครื่องพร็อกซีเซิร์ฟเวอร์เกิดความผิดพลาดในการทำงานทำให้ตัว Squid ซึ่งเป็นโปรแกรมพร็อกซีเซิร์ฟเวอร์หยุดทำงานไปจึงมีผลต่อค่า %Hit และ R ดังที่กล่าวไปแล้ว ในขณะที่รูปที่ 4.3 แสดงให้เห็นถึงค่า S_real ที่วัดได้เปรียบเทียบกับค่า R ซึ่งจากรูปกราฟจะเห็นได้ว่าค่าของ S_real นั้นมีการเพิ่มค่าไปจนถึงค่าประมาณ 18 กิกะไบต์ และจะไม่เพิ่มค่าต่อไป แต่กลับมีการปรับค่าลดลงในบางช่วง

เมื่อพิจารณาดูกราฟค่า R จะเห็นว่าค่า R มีการเปลี่ยนแปลงขึ้นลงอยู่ตลอดเวลาในแต่ละช่วงเวลา ดังนั้นจึงได้นำค่า R ในแต่ละชั่วโมงมาทำการเฉลี่ย โดยทำการเฉลี่ยค่า R ในแต่ละวันแล้วนำมาสร้างกราฟดังแสดงในรูปที่ 4.4 เพื่อดูแนวโน้มของค่า R



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.4 แนวโน้มของค่า R ที่ได้จากค่าเฉลี่ยของค่า R ในรอบ 1 วัน



รูปที่ 4.5 ค่า S_tune, S_real และ S_cal ในช่วงชั่วโมงที่ 80 ถึง 140

และเมื่อนำค่า S_real มาเปรียบเทียบกับค่า S ที่ได้จากการคำนวณ (S_cal) และค่าที่นำไปปรับตั้งจริง (S_tune) มาแสดงเป็นกราฟดังในรูปที่ 4.5 ซึ่งเลือกนำค่า S ในช่วงชั่วโมงที่ 80 ถึง 140 มาแสดงเนื่องจากต้องการขยายให้เห็นถึงค่า S ทั้ง 3 ประเภทได้อย่างชัดเจน ซึ่งจากรูปกราฟจะเห็นได้ว่าค่า S_cal นั้นจะมีค่าสูงสุดอยู่ระหว่างประมาณ 20 ถึง 30 กิกะไบต์ แต่ค่า S_tune จะมีค่ามากที่สุดอยู่ที่ 20 กิกะไบต์ เนื่องจากข้อจำกัดทางด้านเนื้อที่เก็บข้อมูลของเครื่องพร็อกซีเซิร์ฟเวอร์ที่ใช้ในการทดลองและค่า S_real ซึ่งเป็นขนาดของดิสก์แคชจริง จะมีการปรับตัวตามค่า S_tune ที่สั่งปรับค่าในแต่ละครั้ง โดย S_real จะปรับค่าลดลงอย่างรวดเร็วเมื่อ S_tune ที่จะสั่งปรับมีค่าน้อยกว่า S_real แต่ค่า S_real จะค่อยๆ เพิ่มค่าขึ้นอย่างช้า ๆ เมื่อ S_tune มีค่าเพิ่มขึ้นทั้งนี้ขึ้นกับอัตราการร้องขอจากผู้ใช้ที่เข้ามายังพร็อกซีเซิร์ฟเวอร์ในขณะนั้นด้วย และจากรูป 4.5 จะเห็นว่าค่า S_real จะมีค่าคงที่อยู่ที่ประมาณ 18 กิกะไบต์ ทั้งนี้เนื่องจากว่าค่า S_tune ไม่สามารถเพิ่มค่าขึ้นไปได้เกิน 20 กิกะไบต์ทั้งนี้มีสาเหตุเนื่องมาจากกลไกการทำงานภายในของ Squid เองอันได้แก่ การกำหนดค่า cache_swap_high ซึ่งได้กล่าวไว้ในบทที่ 2 และ ข้อจำกัดเกี่ยวกับระบบไฟล์ (File system) ของระบบปฏิบัติการที่ใช้อยู่ที่สามารถจัดเก็บออบเจกต์ได้สูงสุดที่ค่าหนึ่ง

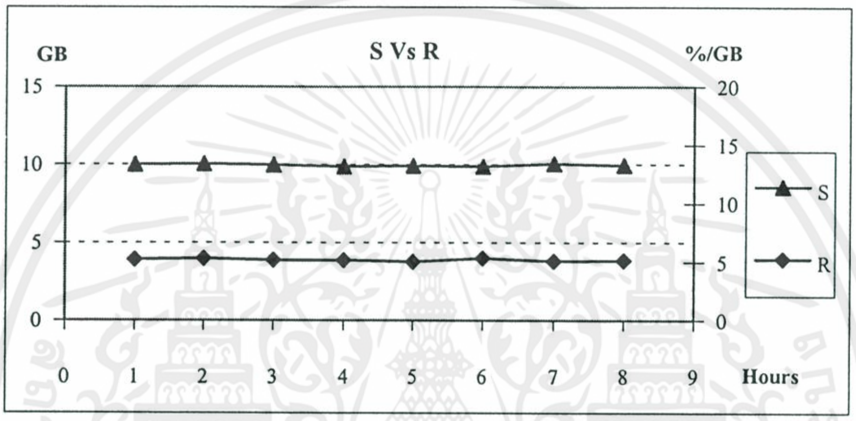
4.3.2 การทดลองปรับค่าอายุอ้างอิง

การทดลองนี้เป็นการทดลองเกี่ยวกับค่าอายุของออบเจกต์ที่อยู่ในดิสก์แคช โดยจะนำค่าเฉลี่ยของอายุที่ได้จากอัลกอริธึมในการวัดอายุของออบเจกต์มาใช้เป็นค่าอายุอ้างอิงในการปรับพารามิเตอร์ reference_age แต่เนื่องจากเครื่อง proxy.kmitl.ac.th ใช้ Replacement policy แบบ LFUDA ซึ่งค่า reference_age จะมีผลต่อ Replacement policy แบบ LRU แต่ไม่มีผลต่อการทำงานแบบ LFUDA จึงเปลี่ยนมาทำการทดลองเปลี่ยนค่าอายุอ้างอิงของออบเจกต์ที่อยู่ในดิสก์แคชของเครื่อง proxy1.kmit.ac.th ซึ่งใช้ Replacement policy แบบ LRU แทนซึ่งเครื่องนี้เป็นเครื่องพร็อกซี

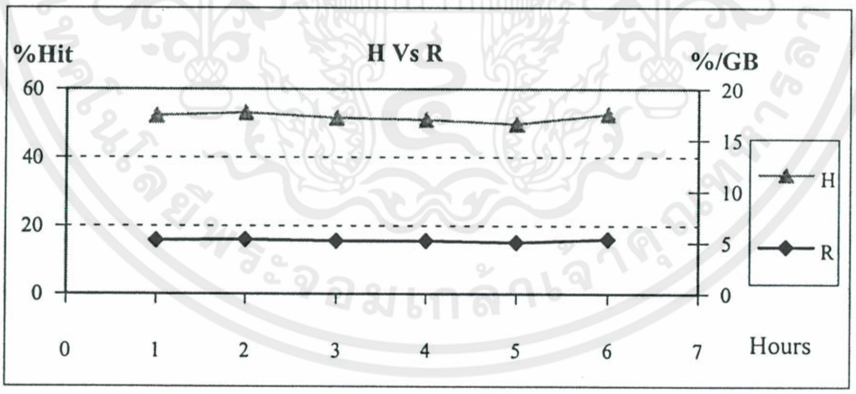
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซิร์ฟเวอร์รองของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง โดยเป็นเครื่อง SUN รุ่น SPARC 2000E มีหน่วยประมวลผลกลาง 4 ตัว หน่วยความจำหลักสำหรับ Squid ในทำพร็อกซีเซิร์ฟเวอร์ 256 เมกะไบต์ และมีพื้นที่ในการทำดิสก์แคช 10 กิกะไบต์โดยเฉพาะ โดยวัดค่า H และ S ทุก 1 ชั่วโมง เมื่อปรับพารามิเตอร์ reference_age ในไฟล์ squid.conf ให้มีค่าเท่ากับ

- 2 สัปดาห์ ซึ่งเป็นค่าที่กำหนดไว้ก่อนจะทำการหาค่าอายุเฉลี่ยของออบเจกต์
- 6 ชั่วโมง ซึ่งได้จากอัลกอริธึมในการคำนวณอายุของออบเจกต์ในบทที่ 3
- 1 ชั่วโมง ซึ่งเป็นหน่วยเวลาที่ต่ำที่สุดที่กำหนดไว้ใน squid.conf



รูปที่ 4.6 ค่า R และ S เมื่อใช้ค่า reference_age เท่ากับ 2 สัปดาห์

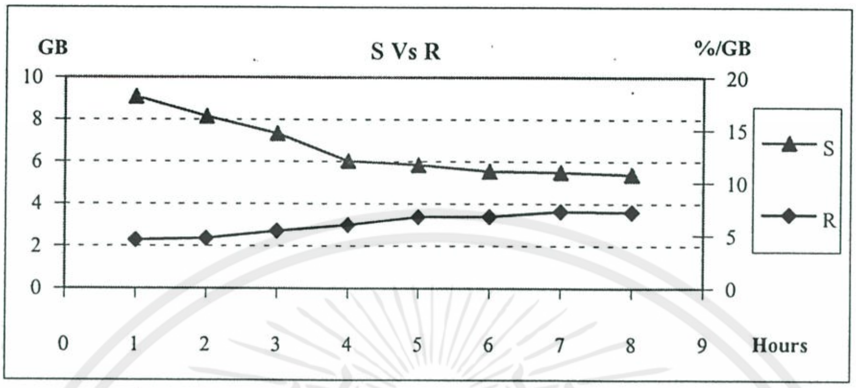


รูปที่ 4.7 ค่า R และ H เมื่อใช้ค่า reference_age เท่ากับ 2 สัปดาห์

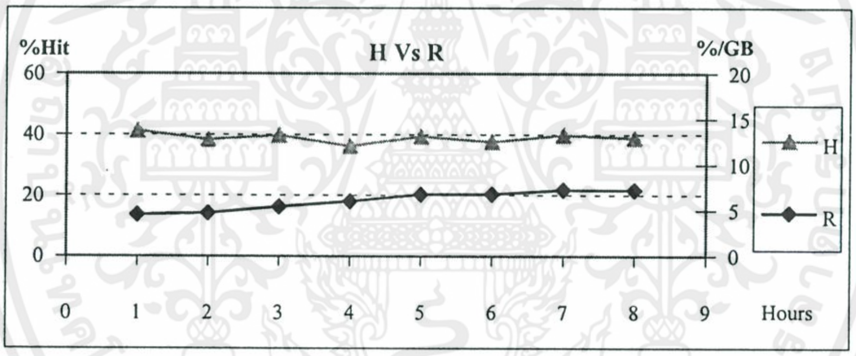
จากกราฟในรูปที่ 4.6 และ 4.7 เป็นการแสดงค่า R ต่อค่า S และ R ต่อค่า H (%Hit) ของเครื่องพร็อกซี proxy1.kmit.ac.th ก่อนที่จะทำการปรับแต่ง โดยมีค่า reference_age เท่ากับ 2 สัปดาห์ซึ่งค่านี้เป็นค่าที่ได้จากการสุ่มเนื่องจากยังไม่มีวิธีการหาค่าที่เหมาะสมมาก่อน

เอกสาร... ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองหาค่าอายุของออบเจกต์ในการทดลองที่ 3.3 ในบทที่ 3ได้ค่าอายุเฉลี่ยของออบเจกต์ออกมาที่ 5 ชั่วโมง 19 นาที จึงทำการเลือกปรับแต่งค่า reference_age เท่ากับ 6 ชั่วโมง และได้ผลดังแสดงในรูปที่ 4.8 และ 4.9



รูปที่ 4.8 ค่า R และ S เมื่อใช้ค่า reference_age เท่ากับ 6 ชั่วโมง

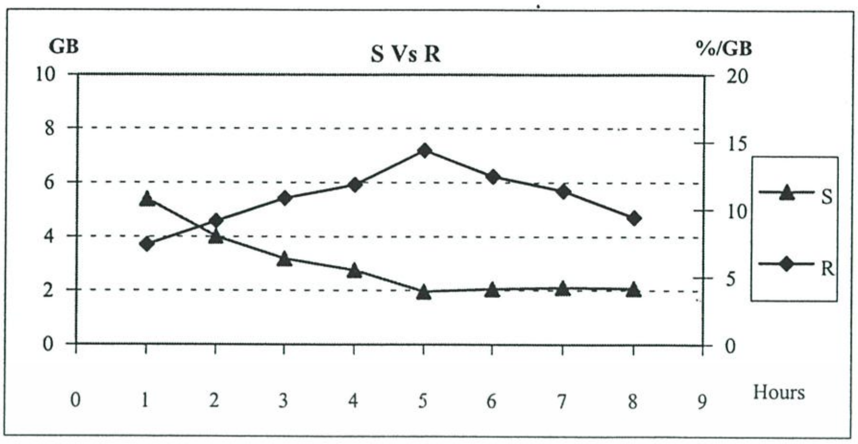


รูปที่ 4.9 ค่า R และ H เมื่อใช้ค่า reference_age เท่ากับ 6 ชั่วโมง

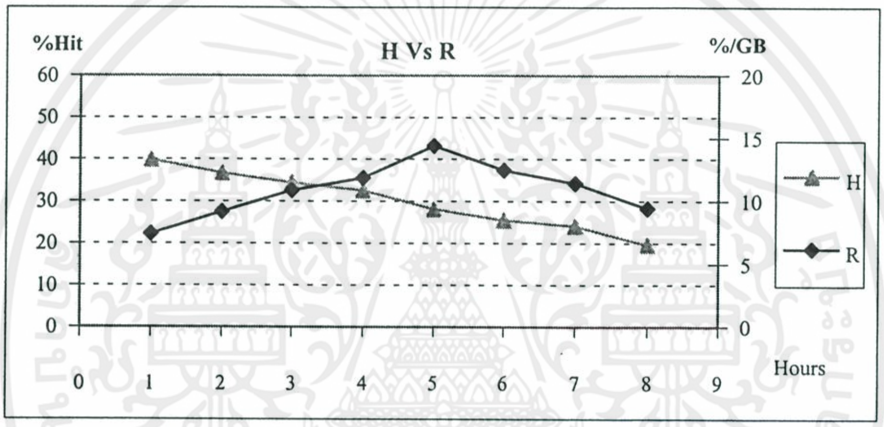
จากรูปที่ 4.8 จะสังเกตเห็นได้ว่าเมื่อเริ่มมีการสั่งปรับค่า reference_age แล้วค่า S จะเริ่มลดลงอย่างต่อเนื่อง ทั้งนี้เนื่องจากมีการลบออบเจกต์ที่มีอายุเกินกว่าที่ตั้งค่าเอาไว้ออกจากแคชและส่งผลให้ค่า R มีค่าสูงขึ้นเนื่องจากค่า S ที่ลดลงในขณะที่ค่า H ค่ามีการเปลี่ยนแปลงน้อยมากดังที่แสดงไว้ในรูปที่ 4.9

จากนั้นทำการปรับลดค่า reference_age เท่ากับ 1 ชั่วโมงซึ่งเป็นค่าต่ำสุดที่ Squid ยอมให้ปรับตั้งได้ในไฟล์ squid.conf และได้ผลดังแสดงในรูปที่ 4.10 และ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



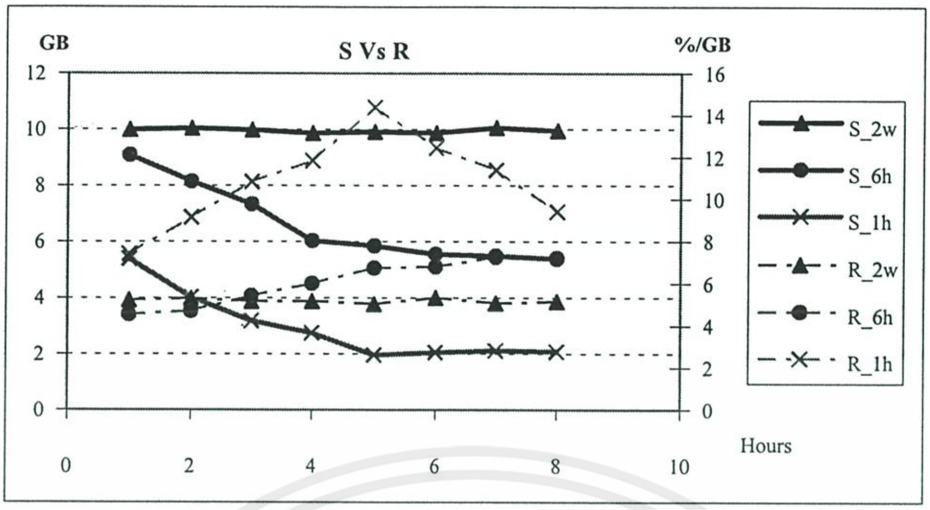
รูปที่ 4.10 ค่า R และ S เมื่อใช้ค่า reference_age เท่ากับ 1 ชั่วโมง



รูปที่ 4.11 ค่า R และ H เมื่อใช้ค่า reference_age เท่ากับ 1 ชั่วโมง

จากรูปที่ 4.10 จะสังเกตเห็นได้ว่าเมื่อเริ่มมีการสั่งปรับค่า reference_age แล้วค่า S ยังคงลดลงอย่างต่อเนื่องเนื่องจากยังมีออบเจกต์ที่มีอายุเกินกว่าที่ตั้งค่าเอาไว้และส่งผลให้ค่า R มีค่าสูงด้วย แต่ค่า S ที่ลดลงมากในระดับที่ต่ำกว่า 3 กิกะไบต์ส่งผลค่า H ลดลงอย่างรวดเร็วดังแสดงไว้ในรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 ค่า S และ R เมื่อใช้ค่า reference_age ต่าง ๆ

จากรูปที่ 4.12 จะเห็นว่าเมื่อทำการปรับค่า reference_age ที่ค่า 2 สัปดาห์ 6 ชั่วโมง และ 1 ชั่วโมงนั้น หลังจากการปรับค่าไปแล้วจะส่งผลให้ค่าขนาดของดิสก์แคช (S) ที่ใช้งานมีค่าลดลงและจะไปค่าที่ที่ค่าหนึ่ง โดยถ้าค่า reference_age มีค่าน้อยลงก็จะทำให้ขนาดของดิสก์แคชน้อยลงด้วยเนื่องจากออบเจกต์ที่มีอายุเกินค่าที่กำหนดจะถูกลบออกไป ในขณะที่ถ้าพิจารณาจากค่า R จะพบว่าการปรับตัวลดลงของค่า S จะส่งผลให้ค่า R มีค่าเพิ่มขึ้น แต่ในกรณีการปรับค่า reference_age ให้เท่ากับ 1 ชั่วโมงนั้นจากรูปกราฟจะเห็นได้ว่าค่า R มีการเพิ่มขึ้นถึงจุดสูงสุดจุดหนึ่งจากนั้นก็จะปรับตัวลดลงอย่างรวดเร็วทั้งนี้เพราะว่าค่า reference_age ที่น้อยทำให้มีออบเจกต์ที่เก็บไว้ในดิสก์แคชน้อยส่งผลให้อัตราการพบข้อมูลในแคช (H) ลดลงอย่างต่อเนื่องทำให้ค่า R มีค่าลดลงตามที่แสดงในรูปกราฟ

4.4 บทวิเคราะห์ผลการปรับแต่ง

ในการทดลองปรับค่าขนาดของดิสก์แคช จะเห็นได้ว่าเราสามารถนำอัลกอริธึมในการปรับหาค่าขนาดของดิสก์แคชตามสมการที่ (4.12) มาใช้เพื่อปรับแต่งพร็อกซีได้อย่างมีประสิทธิภาพโดยผลของการปรับแต่งนั้นจะทำให้ได้ค่า R ที่เพิ่มขึ้นในขณะที่ค่า H มีค่าคงที่ ดังแสดงในรูปที่ 4.4 และจากรูปที่ 4.5 จะเห็นว่าค่าขนาดของดิสก์ที่คำนวณได้ (S_cal) มีค่ามากกว่าที่พื้นที่เก็บข้อมูลบนเครื่องพร็อกซีเซิร์ฟเวอร์จะมีให้ทำให้สรุปได้ว่าขนาดของดิสก์แคชที่ควรจะปรับตั้งนั้นมีค่ามากกว่า 20 กิกะไบต์ขึ้นไป

ในการทดลองปรับค่าอายุอ้างอิง จะเห็นได้ว่าเมื่อทดลองปรับลดค่า reference_age ลงมาจากเดิมที่ตั้งไว้ที่2 สัปดาห์ลงมาเป็น 6 และ 1 ชั่วโมงตามลำดับนั้น ปรากฏว่าได้ค่า R ที่ดีขึ้นคือ ค่า R จะมีค่าเท่ากับ 5.2 6.15 และ 11.75 เปอร์เซ็นต์ต่อกิกะไบต์ แต่ในการปรับค่า reference_age เป็น 1 ชั่วโมงนั้นจะส่งผลให้ค่า H ลดลงอย่างมาก ดังนั้นจึงควรจะใช้ค่าอายุอ้างอิงที่ 6 ชั่วโมงซึ่งเป็นค่า

เฉลี่ยของอายุของออบเจกต์ที่คำนวณได้จากบทที่ 3 สำหรับปรับค่า reference_age ให้กับพร็อกซีเซิร์ฟเวอร์ของเครื่อง proxy1.kmitl.ac.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลงานวิจัยและข้อเสนอแนะ

5.1 สรุปผลงานวิจัยการปรับแต่งค่าขนาดของดิสก์แคชและอายุอ้างอิงสำหรับ Squid พร็อกซีเซิร์ฟเวอร์

จากการศึกษาและวิจัยเกี่ยวกับหลักการทำงานของพร็อกซีเซิร์ฟเวอร์ทำให้สามารถทราบถึงหลักการทำงานและการทำงานร่วมกันของพร็อกซีเซอร์เวอร์ และที่สำคัญคือพารามิเตอร์ในการปรับตั้งค่าพร็อกซีเซิร์ฟเวอร์ที่มีผลต่อการทำงานของเครื่องพร็อกซีเซิร์ฟเวอร์ โดยในงานวิจัยนี้ได้เสนอวิธีการในการคำนวณหาค่าขนาดของดิสก์แคช และค่าอายุอ้างอิง ซึ่งเป็นพารามิเตอร์สำคัญเพื่อนำไปใช้ในการปรับตั้งค่าของพร็อกซีเซิร์ฟเวอร์ โดยได้เลือกทำงานวิจัยกับโปรแกรม Squid ซึ่งเป็นพร็อกซีเซิร์ฟเวอร์ที่ได้รับความนิยมตัวหนึ่ง

ในการคำนวณหาค่าอายุอ้างอิงเพื่อนำไปปรับตั้งนั้นสามารถหาค่าได้จากอัลกอริธึมในการวัดขนาดและอายุของออบเจกต์โดยการนำล็อกไฟล์ซึ่งเป็นผลจากการทำงานของ Squid มาใช้ในการประมวลผล ซึ่งอาศัยหลักการในการจัดประเภทของการร้องขอแต่ละการร้องขอเพื่อที่จะหาการร้องขอประเภทที่สามารถระบุการเปลี่ยนช่วงอายุ (Life MISS, LM) และนำผลต่างช่วงเวลาของการร้องขอประเภทดังกล่าวของแต่ละออบเจกต์มาคำนวณเป็นอายุของออบเจ็กต์แต่ละอัน จากนั้นทำการหาค่าเฉลี่ยของอายุของแต่ละช่วงอายุของแต่ละออบเจกต์ที่คำนวณได้ข้างต้นเพื่อนำไปใช้เป็นค่าอายุอ้างอิงที่จะปรับตั้งให้กับพร็อกซีเซิร์ฟเวอร์ต่อไป ในส่วนของอัลกอริธึมในการวัดอายุของออบเจกต์นั้นยังสามารถจะสามารถระบุถึงความแม่นยำในการวัดอายุของแต่ละออบเจกต์ได้โดยพิจารณาจากจำนวนการร้องขอที่มีเข้ามาในแต่ละช่วงอายุของออบเจกต์นั้นๆ ด้วย โดยถ้ามีจำนวนการร้องขอในแต่ละช่วงอายุมากๆ ก็จะทำให้ค่าอายุที่วัดได้จะมีความแม่นยำใกล้เคียงอายุจริงของออบเจกต์ที่อยู่ในอินเตอร์เน็ทมากขึ้น

นอกจากนี้ยังได้ทำการเสนออัลกอริธึมในการปรับตั้งค่าพารามิเตอร์ที่สำคัญอีกตัวหนึ่งคือขนาดของพื้นที่ที่จะกำหนดให้ Squid นำมาทำเป็นดิสก์แคช โดยอาศัยข้อมูลอัตราการพบข้อมูลในแคชและขนาดของดิสก์แคชที่ใช้จริงซึ่งจะสามารถสะท้อนถึงพฤติกรรมการใช้งานของผู้ใช้เพื่อนำมาใช้ในการคำนวณหาค่าขนาดของดิสก์ที่จะใช้สำหรับปรับตั้งค่า โดยจะทำการหาค่าเฉลี่ยของอัตราการพบข้อมูลและค่าเฉลี่ยขนาดของดิสก์แคชในช่วงเวลาที่ผ่านมาเพื่อนำไปคำนวณหาค่า S

สำหรับที่จะนำไปใช้ปรับตั้งเพื่อให้พร็อกซีเซิร์ฟเวอร์ทำงานในช่วงเวลาครั้งถัดไปตามอัลกอริธึมที่อธิบายไว้ในบทที่ 4

สำหรับการนำอัลกอริธึมในการปรับตั้งค่าอายุอ้างอิงและขนาดของดิสก์แคชไปใช้ในการปรับตั้งกับพร็อกซีเซิร์ฟเวอร์อื่น นั้นสามารถทำได้โดยในกรณีที่พร็อกซีเซิร์ฟเวอร์นั้นเป็น Squid หรือเป็นโปรแกรมพร็อกซีเซิร์ฟเวอร์ที่มีล็อกไฟล์ที่มีรูปแบบเหมือนกันกับล็อกไฟล์ของ Squid ก็จะสามารถนำไปใช้ได้ทันทีเนื่องจากรูปแบบล็อกไฟล์จะเหมือนกัน แต่ถ้าพร็อกซีเซิร์ฟเวอร์นั้นมีรูปแบบของล็อกไฟล์ที่ไม่เหมือนกับของ Squid ก็จำเป็นที่จะต้องมีการปรับรูปแบบในการหาค่าต่างๆ เพื่อที่จะนำมาใช้กับอัลกอริธึมในการวัดขนาดและอายุของออบเจกต์ได้ต่อไป ในส่วนของอัลกอริธึมในการปรับตั้งขนาดของดิสก์แคชนั้นสามารถนำไปใช้ได้ทันทีกับพร็อกซีเซิร์ฟเวอร์อื่นที่สามารถวัดค่าอัตราการพบข้อมูลและค่าเฉลี่ยขนาดของดิสก์แคชได้

จากอัลกอริธึมในการคำนวณหาค่าขนาดของดิสก์แคชที่นำไปปรับค่า(s_{i+1}) สามารถเขียนสรุปเป็นขั้นตอนเพื่อให้สามารถนำไปใช้กับการปรับตั้งพร็อกซีเซิร์ฟเวอร์อื่น ๆ ทั่วไปได้ดังนี้

1. ทำการเก็บค่า H และ s จริงที่เวลาต่างๆ ย้อนหลังไป L ค่าโดยแต่ละค่ามีช่วงห่างที่เท่ากัน และค่า L ควรจะมีค่ามากพอที่จะคลอบคลุมพฤติกรรมการใช้งานในแต่ละวันของผู้ใช้

2. จากนั้นนำค่า H และ s ที่ได้มาทำการประมาณค่าเพื่อหาค่า a และ b ตามสมการ (4.5) โดยในขั้นตอนนี้อาจจะใช้สมการตั้งต้นในการประมาณค่า H ที่ต่างไปจากนี้ได้ ทั้งนี้เนื่องจากพฤติกรรมการใช้งานของแต่ละที่ที่อาจจะไม่เหมือนกัน หรืออาจจะใช้สมการที่ซับซ้อนขึ้นเพื่อการประมาณค่าที่แม่นยำและใกล้เคียงยิ่งขึ้น

3. นำค่า a และ b ใช้ที่คำนวณได้ไปแทนค่าในสมการ (4.11) เพื่อทำการหาค่า $f(s_i)$ ที่เวลาปัจจุบัน

4. จากนั้นจึงทำการคำนวณค่า $f(s_{i-1})$ เพื่อนำมาแทนค่าในสมการ (4.12) เพื่อหาค่า s_{i+1} ที่จะนำไปใช้ในการปรับตั้งค่าดิสก์แคชของ Squid

5. ในกรณีที่ค่า s_{i+1} ที่คำนวณได้มีค่าเกินขอบเขตของขนาดพื้นที่ในการเก็บข้อมูลบนเครื่องพร็อกซีเซิร์ฟเวอร์ก็จะต้องมีการกำหนดให้ค่าขนาดของดิสก์แคชที่จะทำการปรับจริงนั้นอยู่ที่ค่าสูงสุดหรือต่ำสุดที่จะยอมรับได้ โดยนำค่าที่ได้จากการพิจารณาร่วมกับขอบเขตนี้ไปทำการปรับตั้งค่าพารามิเตอร์ cache_dir ในไฟล์ squid.conf

และจากอัลกอริธึมในการปรับค่าอายุอ้างอิงนั้นสามารถสรุปเป็นขั้นตอนได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น

1. ทำการคำนวณหาค่าอายุเฉลี่ยของออบเจกต์ที่ถูกร้องขอผ่านเครื่องพร็อกซีเซิร์ฟเวอร์ที่เราต้องการจะปรับตั้งค่าโดยใช้อัลกอริธึมการวัดอายุของออบเจกต์ที่กล่าวถึงไปแล้วในบทที่ 3 โดยจะต้องมีการจัดเก็บล็อกไฟล์ไว้ก่อนหน้าที่จะทำการคำนวณแล้วในช่วงเวลาที่มากพอที่จะสะท้อนถึง

พฤติกรรมและการเปลี่ยนของออบเจกต์ที่ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์นั้น ๆ ซึ่งการใช้ล็อกไฟล์ขนาดใหญ่นั้นจะทำให้เสียเวลาในการประมวลผลเพื่อวัดค่าอายุของออบเจกต์มากตามไปด้วยดังที่กล่าวไว้ในบทที่ 3 แล้ว

2. นำค่าอายุที่เฉลี่ยของออบเจกต์ที่วัดได้มาทำการปรับตั้งค่าอายุอิงของออบเจกต์ในดิสก์แคช โดยปรับที่พารามิเตอร์ reference_age ในไฟล์ squid.conf ได้โดยตรงทันที

เนื่องจากค่าอายุอ้างอิงจะมีผลต่อการทำงานของพร็อกซีเซิร์ฟเวอร์ที่ใช้ Replacement Policy แบบ LRU ดังนั้นอัลกอริธึมนี้จึงจะใช้ได้ผลในการปรับตั้งพร็อกซีเซิร์ฟเวอร์ที่ใช้ Replancement Policy แบบ LRU และแบบอื่นที่ใช้ค่าอายุอ้างอิงในการตัดสินใจ

ในการนำอัลกอริธึมทั้งสองไปใช้เพื่อปรับตั้งค่าให้กับพร็อกซีเซิร์ฟเวอร์นั้นจะต้องทำการปรับค่าด้วยระยะเวลาที่นานพอสมควรจึงจะทำให้ค่าคำตอบที่ได้จากอัลกอริธึมนั้นมีค่าที่แม่นยำและใกล้เคียงกับพฤติกรรมการใช้งานพร็อกซีเซิร์ฟเวอร์ของผู้ใช้มากขึ้น

5.2 ข้อเสนอแนะของการปรับแต่งค่าพร็อกซีเซิร์ฟเวอร์

ในการทดลองที่ 4.1 นั้นมีการปรับตั้งค่า a, b และ S ที่ถี่มากเนื่องจากข้อจำกัดทางด้านเวลาของผู้ทำการวิจัย ซึ่งในความเป็นจริงแล้วค่า a และ b นั้นน่าจะมีการปรับเพียง 1 ครั้งต่อ 1 วันทั้งนี้เนื่องจากรอบระยะเวลาดังกล่าวจะสามารถคลอบคลุมพฤติกรรมการใช้งานของผู้ใช้ได้ดีกว่าการปรับค่าทุกชั่วโมง นอกจากนี้ในส่วนของค่า R และฟังก์ชันในการประมาณค่า H ตามสมการที่ (4.5) ยังเป็นค่าที่อยู่ในรูปอย่างง่าย (Simple Form) เพื่อที่จะใช้ในการแสดงวิธีการคำนวณค่าขนาดของดิสก์แคช แต่ในกรณีที่ต้องการพัฒนาให้อัลกอริธึมการหาขนาดของดิสก์แคชนี้มีความแม่นยำและใกล้เคียงมากยิ่งขึ้นก็จะสามารถทำได้โดย

- เพิ่มตัวแปรที่เกี่ยวข้องกับราคาค่าใช้จ่ายในการจัดหาพื้นที่ของดิสก์ในค่า R ซึ่งจะทำให้ได้ค่า R ที่ใกล้เคียงกับสภาพของราคาและการเปลี่ยนแปลงของราคาที่แม่นยำมากขึ้น
- ทำงานวิจัยต่อเนื่องเพื่อศึกษาหาฟังก์ชันในการประมาณค่า H ที่ซับซ้อนมากขึ้นซึ่งจะทำให้การประมาณค่า H นั้นทำได้แม่นยำมากขึ้น

ในการทดลองที่ 4.2 นั้นอาจจะต้องทำการทดลองโดยใช้อัลกอริธึมในการวัดอายุของออบเจ็กต์เข้ามาเพื่อหาค่าอายุเฉลี่ยของออบเจกต์ในช่วงเวลา 1 วัน แต่เนื่องจากผู้ทำวิจัยติดปัญหาเรื่องการประมวลผลข้อมูลจากล็อกไฟล์ในแต่ละวันของเครื่องพร็อกซีเซอร์เวอร์ซึ่งในปัจจุบันมีปริมาณข้อมูลสูงขึ้นมาก (ประมาณวันละ400 ถึง 550 เมกะไบต์) จึงไม่สามารถคำนวณค่าอายุเฉลี่ยในแต่ละวันได้ทันเวลา นอกจากจะมีเครื่องที่มีประสิทธิภาพมากกว่าเครื่องที่ใช้ในการทดลองที่ 3.3 มาใช้ใน

การคำนวณ โดยค่าอายุเฉลี่ยที่นำมาใช้ในการทดลองที่สองนี้ได้จากการทดลองตามอัลกอริธึมในการวัดอายุของออบเจกต์จากล็อกไฟล์ที่ได้คำนวณไว้ก่อนหน้านั้นซึ่งได้อธิบายไว้แล้วในบทที่ 3

อนึ่งเนื่องจากการอัลกอริธึมทั้ง 2 อัลกอริธึมที่เสนอมานั้นไม่ได้มีปัจจัยที่นำ Replacement Policy ของดิสก์แคชเข้ามาเกี่ยวข้อง โดยการทดลองปรับตั้งค่าดิสก์แคชในงานวิจัยนี้จะใช้ LFUDA เป็น Replacement Policy เพียงอย่างเดียว จึงอาจจะต้องมีการวิจัยและทดลองเพื่อศึกษาผลกระทบของ Replacement Policy ประเภทต่างๆ ที่อาจจะมีผลต่อการปรับตั้งค่าขนาดของดิสก์เข้ามาเกี่ยวข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] Andrew Cormack, University of Wales, Cardiff "Web Caching" [Online] Available: URL: http://www.jisc.ac.uk/can/caching.html December 2000.

[2] B. Li, M. J.Golin, G. F. Italiano, X. Deng, and K. Sohraby "On the optimal placement of Web proxies in the internet" Proceedings of Infocom'99., 1999

[3] Duane Wessels "Information Resource Caching FAQ" [Online] Available: URL: http://ircache.nlanr.net/Cache/FAQ/ircache-faq.html March 2001.

[4] D.Wessels, K. Claffy, National Laboratory for Applied Network Research "Internet Cache Protocol (ICP), Version 2" RFC 2186, September 1997.

[5] Duane Wessels "Squid Programmers Guide" [Online] Available: URL: http://www.squid-cache.org/Doc/Prog-Guide/prog-guide-2.html March 2001.

[6] Duane Wessels "Squid Cache Server" [Online] Available, URL: http://www.nlanr.net/Squid/index.html March 2001.

[7] Duane Wessels "Squid Frequently Ask Questions" [Online] Available, URL: http://www.squid-cache.org/Doc/FAQ/FAQ-6.html#ss6.6 March 2001.

[8] E. Cohen, B. Krishnamurthy, and J.Rexford "Efficient algorithms for predicting request to Web servers" Proceedings of Infocom'99, 1999.

[9] E. Cohen, B. Krishnamurthy, and J.Rexford, "Evaluating server-assisted cache replacement in the Web" Proceedings European Symposium on Algorithms-98, 1998.

[10] Erwin Kreyszig, "Advanced Engineering Mathematics", 7th Edition, John Wiley & Sons, Inc., 1993. pp.933.

[11] Jeffrey C. Mogul, Compaq Computer Corporation Western Research Laboratory "Squeezing More Bits Out of HTTP Caches", IEEE Network, May/June 2000. pp.6-14.

[12] John Dilley, Martin Arlitt, Stephane Perret, Internet System and Applications Laboratory Hewlett-Packard "Enhancement and Validation of Squid's Cache Replacement Policy" HPL-1999-69 May, 1999.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[13] K. Chinen and S. Yamaguchi "An interactive prefetching proxy server for improvement of WWW latency" Proceedings of INET'97, June 1997.

[14] L. Fan, P. Cao, W. Lin, and Q. Jacobson "Web prefetching between low-bandwitdh clients and proxies: potential and performance" Proceedings of the Sigmetrics'99, 1999.

[15] Martin Arlitt, Ludmila Cherkasova, John Dilley, Richard Friedrich, Tai Jin, Internet System and Applications Laboratory Hewlett-Packard "Evaluating Content Management Techniques for Web Proxy Caches" HPL-98-173 April, 1999.

[16] M. R. Korupolua nd M. Dahlin, "Coordinated placement and replacement for large-scale distributed cache" Proceedings IEEE workshop on Internet Applications, July 1999.

[17] Oskar Pearson "Squid" [Online] Available: URL: http://squid-docs.sourceforge.net/latest/html/book1.htm, March 2001.

[18] Pei Cao, Sandy Irani "Cost-Aware WWW Proxy Caching Algorithms" Proceedings of the 1997 USENIX Symposium on Internet Technology and Systems, 1997. pp. 193-206.

[19] P. Lorenzetti, L. Rizzo, and L. Vicisamo "Replacement policies for a proxy cache" [Online] Available: URL: http://www.iet.unipi.it/luigi/research.html

[20] Rolf Johansson, **"System Modeling and Identification"**, Prentice-Hall International, Inc.,pp. 76, 1993.

[21] Schwartz, Radal L., **"Unix Programming: Learning Perl"**, O'reilly & Associates, Inc., 1994.

[22] S. Williams, M. Abrams, C. R. Standridge, G. Abdulla, and E. A. Fox "Removal policies in network caches for World-Wide Web documents" Proceedings Singcomm'96, 1996.

[23] T. Palpanas and A. Mendelzon "Web prefetching using partial match prediction" Proceedings of WCW'99, 1999.

[24] V. N. Padmanabhan and J. C. Mogul "Using predictive prefetching to improve World Wide Web latency" Proceedings of Sigcomm'96, 1996.

ภาคผนวก

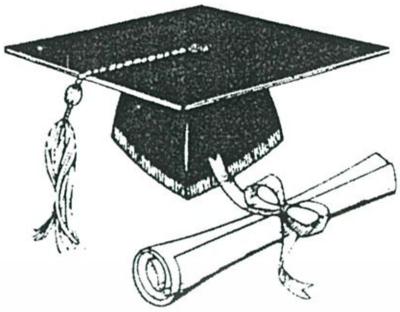
บทความและผลงานวิจัยที่ได้รับการตีพิมพ์

1. วิฑิต อินทรหะ และ อัครินทร์ คุณกิตติ. “การวิเคราะห์ลักษณะของวัตถุใน World Wode Web จาก Squid Log” การประชุมเสนอผลงานวิจัยระดับบัณฑิตศึกษา, ครั้งที่ 1, มิถุนายน 2543.

2. วิฑิต อินทรหะ และ อัครินทร์ คุณกิตติ. “การวัดขนาดและอายุของออบเจกต์ใน World Wide Web จาก Log File” สารเนคเทค, ฉบับที่ 40, 2544.

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



GRAD-RESEARCH

การประชุมเสนอผลงานวิจัยระดับบัณฑิตศึกษา ครั้งที่ 1

First National Symposium on Graduate Research

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัณฑิตวิทยาลัย มหาวิทยาลัยเชียงใหม่

โดยความร่วมมือของที่ประชุมคณบดีบัณฑิตวิทยาลัยมหาวิทยาลัยของรัฐ(ทคบร.)

10-11 มิถุนายน 2543 ณ มหาวิทยาลัยเชียงใหม่

การประชุมเสนอผลงานวิชาการระดับประเทศ

ผลงานวิจัยระดับบัณฑิตศึกษา

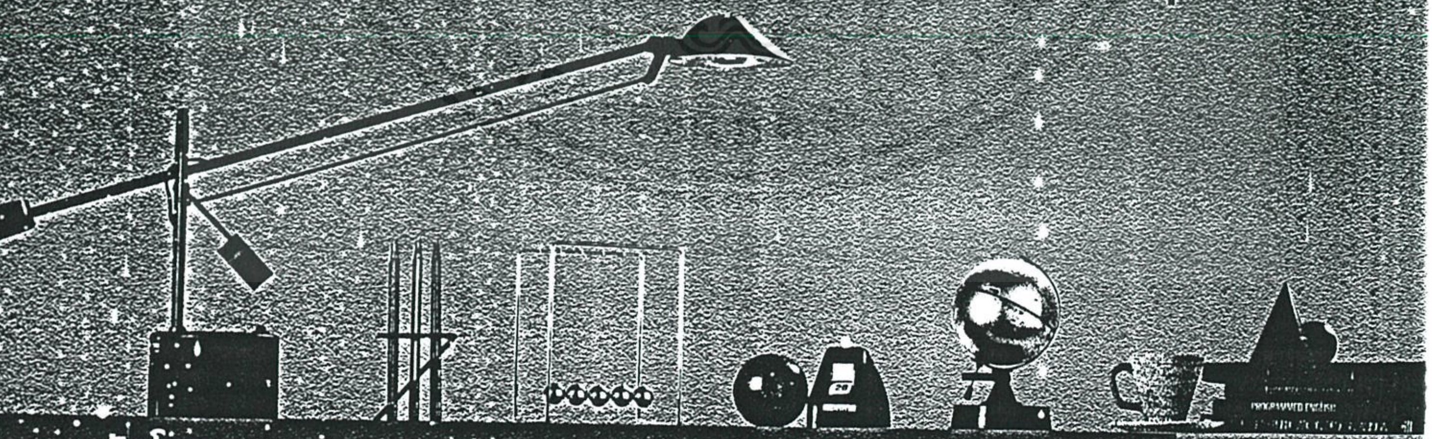
ครั้งที่ 1

First National Symposium on Graduate Research

GRAD-RESEARCH

ณ สำนักบริการวิชาการ มหาวิทยาลัยเชียงใหม่

10-11 มิถุนายน 2543



จัดโดย

บัณฑิตวิทยาลัย มหาวิทยาลัยเชียงใหม่

โดยความร่วมมือของ

ที่ประชุมคณบดีบัณฑิตวิทยาลัยมหาวิทยาลัยของรัฐ (ทคบร.)

ผลงานวิจัยที่ได้รับการตอบรับให้เสนอในการประชุม กลุ่มวิทยาศาสตร์เทคโนโลยี

ST-01	การเตรียมบิสมัธโซเดียมติตาเนตผงละเอียดโดยวิธีทางเคมี	ภูสิต ปุกมณี
ST-02	การพัฒนาระบบวิเคราะห์ที่รวมฟิลด์โฟลแฟรกชันเนชัน และโฟลอินเจคชันอะนาลิซิส	รัตติกาล จันทิวาสน์
ST-03	Mediated Enzyme Electrode for Amperometric Determination of L-glutamate	ปาริยา ณ นคร
ST-04	การหาปริมาณไทเทเนียม (III) และไทเทเนียม (IV) ตามลำดับโดยวิธีโฟลอินเจคชันสเปกโตรโฟโตเมตรี	พัฒนพงษ์ อ่ำพันธุ์
ST-05	ผลของอุณหภูมิแคลไซน์ต่อขนาดอนุภาคของแบเรียมติตาเนตผงละเอียดโดยวิธีออกซาเลต	สามารถ คงทวีเลิศ
ST-06	การปรับปรุงคุณภาพน้ำมันจากเปลือกเมล็ดมะม่วงหิมพานต์เพื่อใช้เป็นเชื้อเพลิงทดแทน	บัณฑิตย์ ศรีสังข์งาม
ST-07	การปรับปรุงวิธีการสกัดและตรวจสอบแบคทีเรียจากดินโดยเทคนิคทางดี เอ็น เอ	ธัญวลี สุขสงวน
ST-08	การศึกษาผลของน้ำสกัดจากเมล็ดมะละกอ *(Carica papaya)* เปรียบเทียบกับฮอร์โมนเอสโตรเจนในหนูขาวที่ตัดรังไข่	สนธยา เนืองศิลป
ST-09	Toxic Algae, *Microcystis aeruginosa* Kutz Distribution in the Kwan Phayao Reservoir Phayao, Thailand	รัฐภูมิ พรหมณะ
ST-10	The Investigation of Toxic Algae in the Reservior of the Mae Kuang Udomtara Dam, Chiang Mai, Thailand	จีรพร เพกเกาะ
ST-11	ผลของน้ำสกัดจากน้ำนมราชสีห์ *(Euphobia hirta Linn.)* ต่อต่อมน้ำนมของหนูขาว	เกศินี ทศวารจันทรา
ST-12	การโคลนเอนไซม์ไคติเนส กลุ่มที่ 1 จากกระถินบ้าน	มานะ ขาวเมฆ
ST-13	การปรับปรุงพันธุ์และการจัดการไรโซเบียมพืชอาหารสัตว์ตระกูลถั่วเพื่อให้มีประสิทธิภาพในการตรึงไนโตรเจนสูงในสภาพไร่นา	กมลลักษณ์ เทียมไธสง
ST-14.	การผลิตและคุณภาพของไวน์ที่ผลิตได้จากองุ่นสายพันธุ์ต่าง ๆ ที่ปลูก ณ ฟาร์มมหาวิทยาลัยเทคโนโลยีสุรนารี	ลำไพ ดิษฐวิบูลย์
ST-15	การใช้เทคโนโลยีชีวภาพเปลี่ยนมันสำปะหลังและวัสดุเหลือทิ้งจากโรงงานแป้งมันสำปะหลังไปเป็นอาหารสัตว์โปรตีนสูง	พจนา ชุ่มขุนทด
ST-16	ความแปรปรวนทางพันธุกรรมของเห็ดอันเนื่องมาจากการถ่ายเชื้ออย่างต่อเนื่องเพื่อการเพาะเห็ด	วรรณกา สัตยาพิสุทธิ์
ST-17	Evaluation of Pesticide Avoidance Behavior by *Anopheles minimus*, a Malaria Vector in Thailand	สังข์สิทธิ์ สังวรณ์โยธิน
ST-18	Determination of the Disulphide Bond within the Loop Connecting α4 and α5 of the *Bacillus thuringiensis* Cry4A toxin	วไลรัตน์ พรพิรุฬห์
ST-19	Double ARMS-PCR Analysis and Confirmation of the Ryanodine Receptor Gene Associated with Malignant Hyperthermia in Swine by a Single Set of Primers	สุวรรณ ช่างกลึงดี
ST-20	การพัฒนาการใช้ DNA Probe เพื่อการแยกเพศปลานิล	ไพ้ภูมิ งามจันทร์
ST-21	The Use of Recombinant Plasmids Carrying *nifA* Gene to Increase Nitrogen Fixation Efficiency in Free-Living *Klebsiella pneumoniae M5a1* and Rice-Rhizobacterium *Klebsiella oxtoca R15*	จักรกฤษฎ์ ภารการ
ST-22	Cloning of Cephalosporin C Acetylesterase from *Bacillus Cereus* BT-24	ทรงศักดิ์ ฤกษ์หริ่ง
ST-23	การเพาะเลี้ยงโปรโตพลาสต์จากใบนมตำเลีย (*Hoya* spp.) และการปลูกถ่ายยีน	สมัชชา นาคสมบัติ
ST-24	การแยกและคัดเลือกเชื้อ เบซิดิโอไมซีท เพื่อใช้ในการฟอกสีเยื่อกระดาษโดยเอนไซม์	นิตยา วันพฤติ
ST-25	ผลของสารสกัดสมุนไพรนมนาง (*Pouteria cambodiana*, Baehni) ในโคนม	ศุภชัย ศรีธิวงศ์
ST-26	การศึกษาเบื้องต้นปริมาณฮอร์โมนเพศ การพัฒนาอวัยวะสืบพันธุ์และอายุของปลาบึก	เกรียงศักดิ์ เม่งอำพัน
ST-27	Study on Irradiation of Flour to Reduce *Bacillus cereus* in Soybean Paste Fermentation	มนชยา ร่มวาปี
ST-28	การศึกษาสมบัติโพรออกซิแดนท์ในเครื่องเทศของไทย	กาญจนา ดำริห์
ST-29	การวิเคราะห์แผ่นฟิล์มเอ็กเรย์เพื่อตรวจหารอยฟันผุโดยใช้วิธีแบบที่ปรับรูปได้	ประจวบ อินระวงศ์
ST-30	การวิเคราะห์ลักษณะของวัตถุใน World Wide Web จาก Squid Log	วิทิต อินทรทะ

การวิเคราะห์ลักษณะของวัตถุใน World Wide Web จาก Squid Log
World Wide Web Object Characteristic Analysis Using Squid Log

วิฑิต อินทรหะ [1], อัครินทร์ คุณกิตติ[2]

[1]นักศึกษาปริญญาโท สาขาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ, [2]อาจารย์
คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ถนนฉลองกรุง เขตลาดกระบัง กทม. 10520

บทคัดย่อ

Squid เป็นพร็อกซี่เซิร์ฟเวอร์ที่ได้รับความนิยมใช้กันเป็นอย่างมากในการที่จะนำมาลดความซ้ำซ้อนและเวลาที่ใช้ในการค้นหาข้อมูล World Wide Web จากเครือข่ายอินเทอร์เน็ต ซึ่งปัจจัยที่มีผลกระทบต่อโอกาสที่จะพบข้อมูลที่ตัวพร็อกซี่เซิร์ฟเวอร์และเวลาเฉลี่ยในการตอบสนองต่อผู้ใช้ของตัวพร็อกซี่เซิร์ฟเวอร์ซึ่งถือเป็นพารามิเตอร์ที่ใช้บ่งชี้ถึงประสิทธิภาพของพร็อกซี่เซิร์ฟเวอร์นั้นมีทั้งในส่วนของพฤติกรรมการร้องขอข้อมูลจากผู้ใช้ และลักษณะของวัตถุที่ถูกร้องขอตลอดจนการปรับตั้งค่าพารามิเตอร์ของพร็อกซี่เซิร์ฟเวอร์เอง งานวิจัยนี้มุ่งศึกษาลักษณะของวัตถุที่ถูกร้องขอในรูปของการกระจายของอายุของวัตถุและการกระจายของขนาดของวัตถุ ซึ่งมีผลกระทบต่อประสิทธิภาพของพร็อกซี่เซิร์ฟเวอร์ โดยจะนำเสนอวิธีการวิเคราะห์ Log File ของ Squid พร็อกซี่เซิร์ฟเวอร์เพื่อทำการหาลักษณะการกระจายของอายุและขนาดของวัตถุ โดยจะทำการศึกษาข้อมูลที่ได้จากพร็อกซี่เซิร์ฟเวอร์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เป็นกรณีศึกษา พร้อมทั้งทำการวิเคราะห์ผลเพื่อหาข้อสรุป

Abstract

Squid is the proxy server which becomes more popular to use in reducing the repetition and time of retrieving data on the internet. User beheavier in requesting data, the characteristic of requested object and how proxy server parameter setting are the important factors which impact on the cache hit rate (%hit) and average response time which are the parameters indicating the performance of proxy 's server. This study will focus on the characteristic of requested object in terms of the distribution of object age and average object size which effect on the parameter of proxy's server. This research will present squid proxy server 's log file Analysis procedure by use KMITL's proxy server 's log file as a case study to analyze the characteristic of the object age and average object size distribution in order to receive the conclusion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

เป็นที่ทราบกันทั่วไปว่าอินเทอร์เน็ตได้เข้ามามีบทบาทในชีวิตประจำวันมากขึ้นทำให้ในปัจจุบันมีผู้ใช้เชื่อมต่อเข้าสู่เครือข่ายนี้เป็นจำนวนมาก ส่งผลให้อัตราการเพิ่มขึ้นของปริมาณข้อมูลที่วิ่งอยู่ในเครือข่ายอินเทอร์เน็ตนั้นเป็นไปอย่างรวดเร็วและต่อเนื่องส่งผลให้เกิดภาวะการคับคั่งของการจราจรของข้อมูลในเครือข่ายอินเทอร์เน็ต โดยเฉพาะที่เส้นทางเชื่อมต่อเข้าสู่อินเทอร์เนทของแต่ละองค์กรจะประสบปัญหาไม่สามารถรองปริมาณความต้องการรับ-ส่งข้อมูลของผู้ใช้ได้อย่างเพียงพอ จึงมีการพัฒนาพร็อกซีเซิร์ฟเวอร์ (Proxy Server) ขึ้นเพื่อนำมาใช้ลดภาระการรับ-ส่งปริมาณข้อมูลของเส้นทางเชื่อมต่อเข้าสู่อินเทอร์เน็ตซึ่งจะเป็นเป็นการประหยัดค่าใช้จ่ายอีกทั้งยังลดเวลาตอบสนองต่อผู้ใช้ลงโดยอาศัยความสามารถในการทำแคช (Cache) ของตัวพร็อกซีเซิร์ฟเวอร์ที่สามารถรองรับการรับ-ส่งข้อมูลด้วย โพรโตคอล HTTP, และ FTP ซึ่งมีความต้องการในการใช้งานกันอย่างสูง ดังนั้นเพื่อให้การใช้งานพร็อกซีเซิร์ฟเวอร์เป็นไปอย่างมีประสิทธิภาพสูงสุดจึงจำเป็นจะต้องทราบถึงลักษณะของวัตถุ (Object) ที่ถูกร้องขอผ่านพร็อกซีเพื่อนำไปใช้ในการปรับแต่งพร็อกซีเซิร์ฟเวอร์ต่อไป

ในงานวิจัยนี้จะอธิบายหลักการทำงานของพร็อกซีเซิร์ฟเวอร์เบื้องต้นเพื่อให้เข้าใจถึงวิธีการในการหาอายุและขนาดของออบเจกต์ในอินเทอร์เน็ตที่ถูกร้องขอจากไคลเอนต์ผ่านทางพร็อกซีเซิร์ฟเวอร์ เพื่อศึกษาลักษณะของออบเจกต์ที่ถูกร้องขอในรูปของการกระจายของอายุและขนาดของออบเจกต์ ซึ่งมีผลกระทบต่อประสิทธิภาพของพร็อกซี่เซิร์ฟเวอร์

หลักการทำงานของพร็อกซีเซิร์ฟเวอร์

การทำงานจะใช้หลักการของ ไคลเอนต์/เซิร์ฟเวอร์ ที่มีตัวเครื่องพร็อกซีเซิร์ฟเวอร์เป็นผู้ให้บริการแก่ ไคลเอนต์ซึ่งก็คือ HTTP ไคลเอนต์ หรือ FTP ไคลเอนต์ โดยบริการในที่นี้หมายถึง การเป็นตัวแทนของไคลเอนต์เหล่านั้นในการไปดึงข้อมูลจากภายนอกมาให้แก่ ไคลเอนต์ตามที่ได้รับการร้องขอมา ทั้งนี้พร็อกซีเซิร์ฟเวอร์ยังสามารถทำการจัดเก็บข้อมูลที่ได้มาเหล่านั้นไว้ชั่วระยะเวลาหนึ่ง (แคช) ซึ่งถ้ามีการร้องขอข้อมูล (ออบเจกต์) เดียวกันเข้ามา พร็อกซีเซิร์ฟเวอร์ก็สามารถนำออบเจกต์ที่จัดเก็บไว้ส่งให้ ไคลเอนต์ได้เลยโดยไม่ต้องไปดึงมาจากภายนอกอีก ในกรณีที่นี้จะเรียกว่าพบ (HIT) ออบเจกต์ในแคช แต่ถ้าออบเจกต์ที่ไคลเอนต์ร้องข้อมาไม่มีอยู่ในแคช หรือกลายเป็นออบเจกต์ใหม่ (ออบเจกต์ที่มีการเปลี่ยนแปลงขนาดและ/หรือวันที่ที่สร้าง) ก็จะเกิดกรณีไม่พบ (MISS) และตัวพร็อกซี่เซิร์ฟเวอร์ก็จะต้องไปดึงข้อมูลจากภายนอกเข้ามาให้กับไคลเอนต์ (Cormack, Andrew,1996)

เมื่อพร็อกซีเซิร์ฟเวอร์ทำงานตามการร้องขอจากไคลเอนต์แต่ละรายการเสร็จก็จะทำการบันทึกผลการทำงานของการร้องขอนั้น ลงสู่ Log File เพื่อแสดงรายละเอียดต่าง ๆ ซึ่งในงานวิจัยนี้จะใช้ไฟล์ชื่อ access.log ของ Squid เวอร์ชัน 2.2 (Wessels, D., 2000) โดยในไฟล์ access.log 1 บรรทัดจะประกอบด้วย 10 ฟิลด์ดังต่อไปนี้ (Wessels, D., 2000)

Timestamp Elapsed Client-Addr Log-Tag/HTTP-Code Size Req-Method URL rfc931 Hierarchy/Hostname Content-Type

ในที่นี้จะขออธิบายความหมายเฉพาะฟิลด์ที่เกี่ยวข้องและนำมาใช้ในงานวิจัยเท่านั้น ได้แก่

Timestamp คือค่าเวลาปัจจุบันที่ Squid จะบันทึกไว้เมื่อเสร็จสิ้นการร้องขอนั้น

Log-Tag/HTTP-Code คือผลการทำงานของ Squid ที่มีต่อการร้องขอนั้น/ผลการทำงานจากโพรโตคอล HTTP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Size คือขนาดของออบเจกต์ที่ไคล์เอนต์ร้องขอ

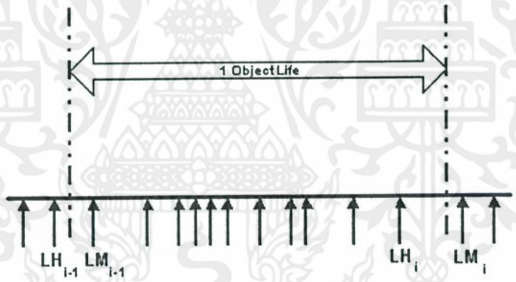
Req.-Method คือวิธีการของการร้องขอเพื่อจะกระทำกับออบเจกต์ เช่น GET, POST เป็นต้น

URL คือ URL ที่ไคลเอนต์ร้องขอ ซึ่งในที่นี้จะใช้เป็นตัวระบุชื่อเพื่อแยกแยะออบเจกต์แต่ละอัน

ดังนั้นในแต่ละบรรทัดของไฟล์ access.log นี้จะเก็บบันทึกข้อมูลที่เกี่ยวกับออบเจกต์ต่างๆ เรียงตามลำดับเวลาก่อนหลังทำให้สามารถนำไฟล์นี้มาใช้ทำการประมวลผลเพื่อหาอายุและขนาดของออบเจกต์ได้

วิธีการหาอายุและขนาดของออบเจกต์จาก Log File

การหาค่าอายุและขนาดของออบเจกต์ (object) ในอินเทอร์เน็ตที่ถูกร้องขอจากผู้ใช้โดยผ่านพร็อกซีจะอาศัยข้อมูลจาก Log File ของโปรแกรมพร็อกซีเซิร์ฟเวอร์ ข้อมูลแต่ละบรรทัดของ Log File คือรายละเอียดของข้อมูลที่แต่ละไคลเอนต์ร้องขอเข้ามายังตัวพร็อกซีเซิร์ฟเวอร์ เมื่อนำข้อมูลจาก Log File มาเขียนแทนด้วยลูกศรบนแกนเวลา ก็จะได้แผนภาพดังแสดงในรูปที่ 1



รูปที่1 หลักการหาอายุของออบเจกต์จากข้อมูลการร้องขอข้อมูลในพร็อกซีเซิร์ฟเวอร์

จากรูปที่ 1 ซึ่งมีการร้องขอเข้ามาทางซ้ายสุดซึ่งอาจจะเป็นการร้องขอแรกที่บันทึกไว้ในไฟล์และต่อมาเมื่อมีการร้องขอออบเจกต์ที่เราสนใจนี้เข้ามาอีกและออบเจกต์นั้นถูกพบในแคชของพร็อกซีเซิร์ฟเวอร์และพร็อกซีเซิร์ฟเวอร์ตรวจสอบแล้วว่าออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตนั้นยังไม่มีการเปลี่ยนแปลง เราจะจัดให้การร้องขอนี้อยู่ในประเภทของการร้องขอแบบ Life_Hit (LH) และต่อมาเมื่อมีการร้องขอออบเจกต์นี้เข้ามาอีกแต่ออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตมีการเปลี่ยนแปลงพร็อกซี่เซิร์ฟเวอร์ก็จะต้องไปเอาออบเจกต์นี้มาเก็บลงในแคชแทนออบเจกต์ตัวเดิมที่มีอยู่ก่อนหน้า ในกรณีนี้เราจะจัดให้การร้องขอนี้อยู่ในประเภทของการร้องขอแบบ Life_Miss (LM) ซึ่งในความเป็นจริงการร้องขอที่เข้ามายังพร็อกซี่เซิร์ฟเวอร์นั้นอาจจะมีการร้องขอออบเจกต์เดียวกันจากหลาย ๆ ไคล์เอนท์ในเวลาไล่เลี่ยกัน จึงมีโอกาสของการเกิดกรณีของ Life_Hit มากกว่า Life_Miss นอกจากนี้ยังอาจจะมีการร้องขอที่ยังไม่สามารถระบุได้แน่ชัดว่าออบเจกต์มีการเปลี่ยนแปลงหรือไม่ซึ่งเราจะจัดให้อยู่ในประเภทของการร้องขอแบบ Life_Unknown (LU) ส่วนกรณีของการร้องขอที่ไม่สำเร็จหรือไม่มีผลต่อการหาอายุและขนาดของออบเจกต์จะจัดอยู่ในการร้องขอแบบ Life_Ignore (LI)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนำข้อมูลในแต่ละบรรทัดของ Log File มาใช้ทำการจัดแบ่งประเภทของการร้องขอที่เกิดขึ้นโดยออกเป็น 4 ประเภทได้แก่ Life_Miss, Life_Hit, Life_Unknown และ Life_Ignore โดยจะใช้ข้อมูลในฟิลด์ที่ 4 ของไฟล์ access.log ในส่วนของ *Log-Tag* ของแต่ละบรรทัดมาเป็นเกณฑ์ในการจัดแบ่งประเภทดังแสดงในตารางที่ 1

ตารางที่ 1 ประเภทของ การร้องขอ ที่มีผลต่อการหาอายุของออบเจกต์

ประเภท	ประเภทของการร้องขอ (Request Type)	รายละเอียด
Life_Miss (LM)	TCP_REFRESH_MISS	คือ การร้องขอที่เกิดเมื่อ object มีการเปลี่ยนแปลงอายุหรือวันที่สร้าง
Life_Hit (LH)	TCP_REFRESH_HIT,TCP_IMS_HIT, TCP_IMS_MISS	เป็น การร้องขอที่เกิดเมื่อ object ไม่มีการเปลี่ยนแปลงอายุหรือวันที่สร้าง
Life_Unknown (LU)	TCP_HIT, TCP_MISS, TCP_MEM_HIT, TCP_CLIENT_REFRESH_MISS, UDP_HIT, UDP_MISS	เป็น การร้องขอที่ยังไม่สามารถชี้ชัดได้ object มีการเปลี่ยนแปลงหรือไม่ จำเป็นต้องดูการเปลี่ยนแปลงขนาดและวันที่สร้างประกอบ
Life_Ignore (LI)	NONE, ERR_*, TCP_DENIDE, TCP_NEGATIVE_HIT,TCP_SWAPFAIL	เป็น การร้องขอที่ไม่เกี่ยวข้องกับการเปลี่ยนแปลง object ซึ่งจะไม่นำมาประมวลผล

ประเภทของการร้องขอที่จะนำมาใช้ในการคำนวณหาอายุของออบเจกต์นั้นได้แก่ประเภท Life_Miss, Life_Hit และ Life_Unknown ซึ่งในสองประเภทแรกสามารถนำไปคำนวณได้เลย แต่ในกรณีของ Life_Unknown นั้นจะต้องมีการตรวจสอบก่อนว่าการร้องขอที่ทำให้เกิดกรณีของ Life_Unknown นั้นมีการเปลี่ยนแปลงขนาดหรือไม่ ถ้ามีการเปลี่ยนแปลงขนาดจึงจะถือว่าออบเจกต์นั้นมีการเปลี่ยนแปลงและจะอนุโลมให้การร้องขอนี้จัดอยู่ในประเภท Life_Miss เพื่อนำไปใช้ในการคำนวณต่อไป

เราสามารถประมาณอายุของออบเจกต์ในช่วงที่ i ได้จากการนำเวลาของการเกิด Life_Miss ครั้งล่าสุด(LM_i) และ Life_Hit ครั้งล่าสุด(LH_i)มาหาค่าเฉลี่ยแล้วลบด้วยค่าเฉลี่ยของเวลาที่เกิดการเปลี่ยนแปลงของออบเจกต์ในช่วงก่อนหน้านั้น(LM_{i-1} , LH_{i-1}) ดังแสดงในสมการ

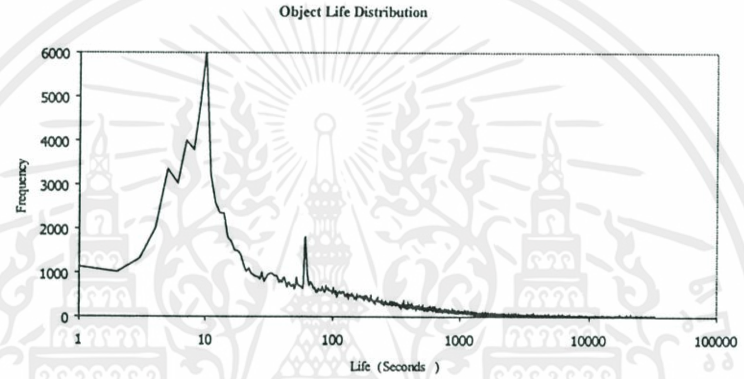
$$objectlife_i = \frac{(LM_i + LH_i)}{2} - \frac{(LM_{i-1} + LH_{i-1})}{2} \quad \text{.................(1)}$$

ในส่วนของขนาดของออบเจกต์นั้นเราสามารถหาได้จากฟิลด์ที่ 5 จากไฟล์ access.log โดยมีข้อแม้ว่าเราไม่ได้เก็บค่าขนาดของออบเจกต์ทุกๆ ค่าที่ปรากฏอยู่ในไฟล์นั้น แต่จะเก็บค่าขนาดของออบเจกต์ก็ต่อเมื่อพบว่าออบเจกต์นั้นๆ มีการเปลี่ยนแปลงเกิดขึ้นเท่านั้น (ในกรณีที่เกิดการร้องขอที่จัดอยู่ในประเภท Life_Miss) โดยจะนับจำนวนของอายุของออบเจกต์ไปด้วย

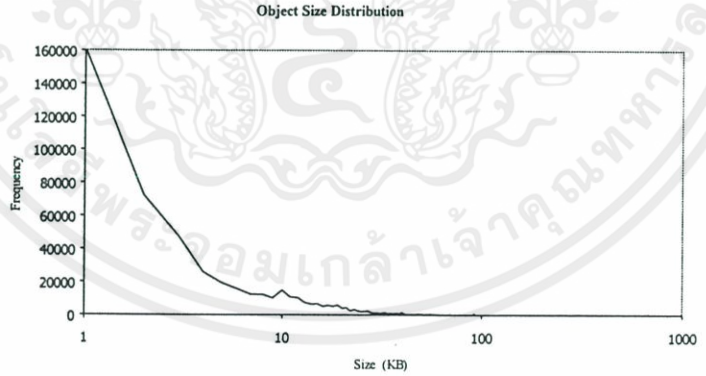
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลอง

จากวิธีการหาอายุและขนาดของออบเจกต์ข้างต้นได้นำมาพัฒนาโปรแกรมโดยใช้ Perl Script (Schwartz, Radal L., 1994) เพื่อช่วยในการประมวลผล ซึ่งในการทดลองนั้นจะใช้ Perl Script ที่ได้มาประมวลผล Log File จาก เครื่อง proxy1.kmitl.ac.th ซึ่งทำหน้าที่เป็นพร็อกซี่เซิร์ฟเวอร์หลักของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง ในช่วงเวลาตั้งแต่วันที่ 1 ถึง 14 กุมภาพันธ์ พ.ศ.2543 ซึ่งคิดเป็นข้อมูลการร้องขอ 9,422,162 requests (เฉลี่ยวันละ 673,011 requests) มาทำการหาการกระจายของขนาดและอายุของออบเจกต์โดย ได้ผลดังแสดงในรูปที่ 3 และ 4



รูปที่ 3 กราฟแสดงการกระจายของขนาดของออบเจกต์



รูปที่ 4 กราฟแสดงการกระจายของอายุของออบเจกต์

ผลการทดลองแสดงให้เห็นถึงการกระจายตัวของขนาดของออบเจกต์ส่วนใหญ่จะมีค่าไม่เกิน 10 กิโลไบต์ โดยมีค่าเฉลี่ยอยู่ที่ 8.78 กิโลไบต์ ในส่วนการกระจายอายุของออบเจกต์ส่วนมากมีค่าน้อยกว่า 10 วินาทีโดยมีค่าเฉลี่ยอยู่ที่ 19128.29 วินาที (ประมาณ 5 ชั่วโมง 19 นาที) ทั้งนี้มีออบเจกต์จำนวน 347,440 ออบเจกต์ที่มีขนาดอยู่ระหว่าง 0 ถึง 1 กิโลไบต์ และอีก 885 ออบเจกต์ที่มีอายุอยู่ระหว่าง 0 ถึง 1 วินาที เนื่องจากในความเป็นจริงมีออบเจกต์บางตัวที่มีขนาดไม่ถึง 1 กิโลไบต์ (เช่น มีขนาด 250 ไบต์) และ/หรือ มีอายุไม่ถึง 1 วินาที

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุป

จากงานวิจัยนี้เราสามารถหาทั้งขนาดและอายุของวัตถุที่ถูกไคลเอนต์ร้องขอจากอินเทอร์เน็ตผ่านทางพร็อกซีเซิร์ฟเวอร์ได้จาก Log File ของพร็อกซีเซิร์ฟเวอร์ โดยไม่จำเป็นต้องศึกษาจากออบเจกต์ทั้งหมดที่มีอยู่ในอินเทอร์เน็ต เนื่องจากลักษณะของออบเจกต์ที่ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์จะสะท้อนถึงพฤติกรรมการใช้งานของผู้ใช้และลักษณะของออบเจกต์ในอินเทอร์เน็ตซึ่งเราสามารถนำไปใช้เป็นข้อมูลในการปรับแต่งประสิทธิภาพของพร็อกซีได้ต่อไป

เอกสารอ้างอิง

Cormack, Andrew, 1996. Web Caching,[Online] Available, URL: http://www.jisc.ac.uk/acn/caching.html

Wessels, D., 2000, Squid Cache Server, [Online] Available, URL: http://www.nlanr.net/Squid/index.html

Wessels, D., 2000, Squid Frequently Ask Questions, [Online] Available,
URL: http://www.squid-cache.org/Doc/FAQ/FAQ-6.html#ss6.6

Schwartz, Radal L., 1994, Unix Programming: Learning Perl, O'reilly & Associates, Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ECTI-21 NECTEC
ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ
National Electronics and Computer Technology Center

ECTI: Electronic*Computer*Telecommunication*Information

สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ
73/1 ถนนพระราม 6 เขตราชเทวี กรุงเทพฯ 10400
โทร. (662) 644-8150-90 โทรสาร. (662) 644-8137-8

National Science and Technology Development Agency
73/1 Rama VI rd., Ratchathewi, Bangkok 10400, THAILAND.
Tel. (662) 644-8150-90 Fax. (662) 644-8137-8

ที่ วว. 5204.0104/017

๗ กุมภาพันธ์ ๒๕๔๔

เรื่อง การตอบรับบทความ

เรียน คุณ วิทิต อินทรหะ

เนื่องด้วยศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ ได้มอบหมายให้ งานประชาสัมพันธ์และมัลติมีเดีย จัดทำวารสาร "สารเนคเทค" ขึ้น เพื่อใช้เป็นเวทีในเชิงวิชาการและเผยแพร่ข้อมูลข่าวสารด้านคอมพิวเตอร์ เทคโนโลยีสารสนเทศ แก่บุคคลทั่วไปที่สนใจ

ในการนี้ทางกองบรรณาธิการจึงเห็นสมควรนำบทความของท่าน นักศึกษาปริญญาโท คณะเทคโนโลยีสารสนเทศ สาขาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ชื่อ "การวัดขนาดและอายุออบเจ็กต์ใน World Wide Web จาก Log File" (World Wide Web Object Size and Life Measurement from Log File) ลงตีพิมพ์ในวารสารฉบับที่ ๔๐ เดือนพฤษภาคม-เมษายน ๒๕๔๔

จึงเรียนมาเพื่อทราบ

ขอแสดงความนับถือ

(ลัญจนา นิตยพัฒน์)
หัวหน้างานประชาสัมพันธ์และมัลติมีเดีย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานประชาสัมพันธ์และมัลติมีเดีย
โทรศัพท์ 644-8150-9 ต่อ 725
โทรสาร 644-8129

การวัดขนาดและอายุของออบเจกต์ใน World Wide Web จาก Log File

World Wide Web Object Size and Life Measurement from Log File

วิฑิต อินทรหะ [1], อัครินทร์ คุณกิตติ[2]

[1]นักศึกษาปริญญาโท สาขาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ, [2]อาจารย์

คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ถนนฉลองกรุง เขตลาดกระบัง กทม. 10520

บทคัดย่อ

ขนาดและอายุของออบเจกต์ในอินเทอร์เน็ตที่ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์เป็นพารามิเตอร์สำคัญที่จะนำไปใช้ในการปรับแต่งประสิทธิภาพในการทำงานของพร็อกซีเซิร์ฟเวอร์เพื่อให้ทำงานได้อย่างมีประสิทธิภาพ เราสามารถหาขนาดของออบเจกต์ที่ถูกร้องขอได้โดยตรงจาก Log file ของ Squid แต่ไม่สามารถจะทราบค่าอายุที่แท้จริงของทุก ๆ ออบเจกต์ที่มีอยู่ได้โดยตรง ในบทความนี้จึงได้เสนอการวัดค่าอายุของออบเจกต์โดยใช้หลักการในการหาประเภทของการร้องขอที่บ่งบอกถึงการเปลี่ยนช่วงอายุจาก Squid Log เพื่อทำการคำนวณวัดค่าอายุ พร้อมทั้งหาค่าขอบเขตความผิดพลาดจากการคำนวณอายุดังกล่าวด้วย จากผลการทดลองพบว่าวิธีการคำนวณที่เสนอมีความผิดพลาดในการวัดอยู่ในขอบเขตที่ได้จากการวิเคราะห์ และจะมีค่าลดลงเมื่ออัตราการร้องขอมีความถี่เพิ่มขึ้น จากนั้นจึงได้ทำการทดลองวัดค่าอายุและค่าความผิดพลาดที่เกิดจากการคำนวณอายุจาก Log file จากเครื่องพร็อกซีเซิร์ฟเวอร์ของสถาบันเทคโนโลยีเจ้าคุณทหาร ลาดกระบัง อีกด้วย

Abstract

Size and life of the object, which is requested from the Internet via proxy server, are the key parameter in order to tune up proxy server 's performance. We can determine object's size directly from the Squid's Log File. However, we are not able to identify each object's life precisely. This article will present the calculation of object 's life by deploying the concept of analyze object's request which indicates the life change from the Squid's Log File in order to determine the life measurement and error boundary from the calculation. From the experiment, we found that the errors from this measurement are within the predicted boundary and the increment of the request rate will reduce these errors. In addition, we have also finished the experiment of life measurement and the errors, which are from the Log File of KMITL's proxy server

บทนำ

สืบเนื่องมากจากการการขยายตัวอย่างรวดเร็วของเครือข่ายอินเทอร์เน็ต (Internet) ซึ่งมีจำนวนผู้ใช้และปริมาณการรับส่งข้อมูลผ่านเครือข่ายนี้เป็นจำนวนมากและเพิ่มขึ้นอย่างต่อเนื่อง จึงมีการนำพร็อกซีเซิร์ฟเวอร์ (Proxy Server) มาอย่างแพร่หลายเพื่อประโยชน์ในการประหยัดแบนด์วิดธ์ (Bandwidth) ของเส้นทางการเชื่อมต่อเข้าสู่อินเทอร์เน็ตและการลดเวลาในการตอบสนองต่อการร้องขอข้อมูลจากผู้ใช้ ทางคณะผู้ทำวิจัยที่ได้ทำการศึกษาหลักการทำงานของพร็อกซีเซิร์ฟเวอร์และรวมไปถึงลักษณะของออบเจกต์ (Object) ที่มีอยู่ในอินเทอร์เน็ตซึ่งถูกร้องขอจากผู้ใช้ผ่าน

... เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ... และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางพร็อกซีเซิร์ฟเวอร์ โดยได้มุ่งศึกษาขนาดและอายุของออบเจกต์ที่ถูกร้องขอผ่านพร็อกซีเซิร์ฟเวอร์เพื่อนำไปใช้เป็นข้อมูลในการปรับแต่งการทำงานของพร็อกซีเซิร์ฟเวอร์ให้ทำงานได้อย่างมีประสิทธิภาพต่อไป

ในงานวิจัยนี้จะอธิบายถึงวิธีการหาอายุและขนาดของออบเจกต์ในอินเทอร์เน็ตที่ถูกไคลเอนต์ร้องขอผ่านพร็อกซีเซิร์ฟเวอร์จากล็อกไฟล์ (Log File) ของ พร็อกซีเซิร์ฟเวอร์ เพื่อศึกษาลักษณะของออบเจกต์ที่ถูกร้องขอในรูปของการกระจายของอายุและขนาดของออบเจกต์ และยังได้แสดงค่าความผิดพลาดที่เกิดจากวิธีการที่ใช้ในการหาอายุของออบเจกต์ที่ถูกร้องขอตามวิธีข้างต้น

หลักการทำงานของพร็อกซีเซิร์ฟเวอร์

การทำงานของระบบที่มีพร็อกซีเซิร์ฟเวอร์ติดตั้งอยู่จะใช้หลักการของ ไคลเอนต์/เซิร์ฟเวอร์ (Client/Server) ที่มีตัวเครื่องพร็อกซีเซิร์ฟเวอร์เป็นผู้ให้บริการแก่ไคลเอนต์ซึ่งในที่นี้คือ HTTP ไคลเอนต์ โดยบริการในที่นี้หมายถึงการเป็นตัวแทนของไคลเอนต์เหล่านั้นในการไปเรียกข้อมูลจากเครื่องเว็บเซิร์ฟเวอร์ (Web Server) มาให้แก่ไคลเอนต์ตามที่ได้รับการร้องขอมา ทั้งนี้พร็อกซีเซิร์ฟเวอร์ยังสามารถทำการจัดเก็บข้อมูลที่ได้มาเหล่านั้นไว้ชั่วระยะเวลาหนึ่ง (Cache) ซึ่งถ้ามีการร้องขอข้อมูล (ออบเจกต์) เดียวกันเข้ามา พร็อกซีเซิร์ฟเวอร์ก็สามารถนำออบเจกต์ที่จัดเก็บไว้ส่งให้ ไคลเอนต์ได้เลยโดยไม่ต้องไปดึงมาจากภายนอกอีก ในกรณีที่นี้จะเรียกว่าพบ (HIT) ออบเจกต์ในแคช (Cache) แต่ถ้าออบเจกต์ที่ไคลเอนต์ร้องข้อมาไม่มีอยู่ในแคชหรือกลายเป็นออบเจกต์ใหม่ (เกิดจากกรณีที่ออบเจกต์มีการเปลี่ยนแปลงขนาดและ/หรือวันที่ที่สร้าง) ก็จะเกิดกรณีไม่พบ (MISS) และตัวพร็อกซี่เซิร์ฟเวอร์ก็จะต้องไปดึง ออบเจกต์จากภายนอกเข้ามาให้กับไคลเอนต์ [1]

เมื่อพร็อกซีเซิร์ฟเวอร์ทำงานตามการร้องขอจากไคลเอนต์แต่ละรายการเสร็จก็จะทำการบันทึกผลการทำงานของการร้องขอนั้น ลงสู่ Log File เพื่อแสดงรายละเอียดต่างๆ ซึ่งในงานวิจัยนี้จะใช้ไฟล์ชื่อ access.log ของ Squid เวอร์ชัน 2.2 [2] โดยในไฟล์ access.log 1 บรรทัดจะประกอบด้วย 10 ฟิลด์ดังต่อไปนี้ [3]

Timestamp Elapsed Client-Addr Log-Tag/HTTP-Code Size Req-Method URL rfc931 Hierarchy/Hostname Content-Type

ในที่นี้จะขออธิบายความหมายเฉพาะฟิลด์ที่เกี่ยวข้องและนำมาใช้ในงานวิจัยเท่านั้น ได้แก่

Timestamp คือค่าเวลาปัจจุบันที่ Squid จะบันทึกไว้เมื่อเสร็จสิ้นการร้องขอนั้น

Log-Tag/HTTP-Code คือผลการทำงานของ Squid ที่มีต่อการร้องขอนั้น/ผลการทำงานจากโพรโตคอล HTTP

Size คือขนาดของออบเจกต์ที่ไคล์เอนต์ร้องขอ

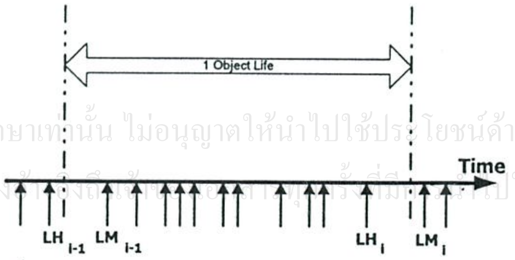
Req.-Method คือวิธีการของการร้องขอเพื่อจะกระทำกับออบเจกต์ เช่น GET, POST เป็นต้น

URL คือ URL ที่ไคลเอนต์ร้องขอ ซึ่งในที่นี้จะใช้เป็นตัวระบุชื่อเพื่อแยกแยะออบเจกต์แต่ละอัน

ดังนั้นในแต่ละบรรทัดของไฟล์ access.log นี้จะเก็บบันทึกข้อมูลที่เกี่ยวกับออบเจกต์ต่างๆ เรียงตามลำดับเวลาก่อนหลังทำให้สามารถนำไฟล์นี้มาใช้ทำการประมวลผลเพื่อหาอายุและขนาดของออบเจกต์ได้

วิธีการหาอายุและขนาดของออบเจกต์จาก Log File

การหาค่าอายุและขนาดของออบเจกต์ (object) ในอินเทอร์เน็ตที่ถูกร้องขอจากผู้ใช้โดยผ่านพร็อกซีจะอาศัยข้อมูลจาก Log File ของโปรแกรมพร็อกซีเซิร์ฟเวอร์ ข้อมูลแต่ละบรรทัดของ Log File คือรายละเอียดของข้อมูลที่แต่ละไคลเอนต์ร้องขอเข้ามายังตัวพร็อกซีเซิร์ฟเวอร์ เมื่อนำข้อมูลจาก Log File มาเขียนแทนด้วยลูกศรบนแกนเวลา ก็จะได้แผนภาพดังแสดงในรูปที่ 1



รูปที่1 หลักการหาอายุของออบเจกต์จากข้อมูลการร้องขอข้อมูลในพร็อกซีเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

จากรูปที่1 มีการร้องขอออบเจกต์เข้ามาทางซ้ายสุดของแกนเวลาซึ่งอาจจะเป็นการร้องขอแรกที่บันทึกไว้ในล็อกไฟล์และต่อมาเมื่อมีการร้องขอออบเจกต์ที่เราสนใจนี้เข้ามาอีกและพร็อกซีเซิร์ฟเวอร์พบ ออบเจกต์นั้นในแคชของและได้ทำการตรวจสอบแล้วว่าออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตนั้นยังไม่มีการเปลี่ยนแปลงขนาดและ/หรือ วันที่ออบเจกต์นั้นถูกสร้าง จึงได้ทำการนำออบเจกต์นั้นที่เก็บไว้ในแคชส่งไปให้ไคลเอนท์ที่ร้องขอมา เราจะจัดให้การร้องขอดังกล่าวนี้อยู่ในประเภทของการร้องขอแบบ Life_Hit (LH) จากนั้นการร้องขอครั้งถัดไปดังแสดงในรูปเมื่อมีการร้องขอออบเจกต์นี้เข้ามาอีกแต่ออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตมีการเปลี่ยนแปลงพร็อกซีเซิร์ฟเวอร์ก็จะต้องไปเอาออบเจกต์นี้มาเก็บลงในแคชแทนออบเจกต์ตัวเดิมที่มีอยู่ก่อนหน้านี้และทำการส่งออบเจกต์นั้นให้กับไคลเอนท์ ในกรณีนี้เราจะจัดให้การร้องขอดังกล่าวอยู่ในประเภทของการร้องขอแบบ Life_Miss (LM) ซึ่งในความเป็นจริงการร้องขอที่เข้ามายังพร็อกซีเซิร์ฟเวอร์นั้นอาจจะมีการร้องขอออบเจกต์เดียวกันจากหลาย ๆ ไคลเอนท์ในเวลาไล่เลี่ยกัน จึงมีโอกาสของการเกิดกรณีของ Life_Hit มากกว่า Life_Miss นอกจากนี้ยังอาจจะมีผลของการร้องขอที่ยังไม่สามารถระบุได้แน่ชัดว่าออบเจกต์มีการเปลี่ยนแปลงหรือไม่ซึ่งเราจะจัดให้อยู่ในประเภทของการร้องขอแบบ Life_Unknown (LU) ส่วนกรณีของการร้องขอที่ไม่สำเร็จหรือไม่มีผลต่อการหาอายุและขนาดของออบเจกต์จะจัดอยู่ในการร้องขอแบบ Life_Ignore(LI) ซึ่งเราจะไม่นำมาใช้ในการคำนวนหาอายุและขนาดของออบเจกต์

เราสามารถที่จะจัดแยกประเภทของการร้องขอแต่ละครั้งเพื่อนำไปใช้ในการคำนวนหาอายุของออบเจกต์ ได้จากข้อมูลผลของการร้องขอซึ่งเป็นข้อมูลในฟิลด์ที่ 4 ในส่วนของ *Log-Tag* ของไฟล์ access.log โดยได้จัดแยกประเภทของการร้องขอตามผลของการร้องขอดังแสดงในตารางที่ 1 ซึ่งจะแสดงประเภทของการร้องขอ (Request Type) ซึ่งเป็นข้อมูลในฟิลด์ที่ 4 ของไฟล์ access.logในส่วนของ *Log-Tag* ของแต่ละบรรทัด ที่นำมาเป็นเกณฑ์ในการจัดแบ่งประเภทดังแสดงในตารางที่ 1

ตารางที่ 1 ประเภทของ การร้องขอ ที่มีผลต่อการหาอายุของออบเจกต์

ประเภท	ผลของการร้องขอที่ปรากฎใน Log-Tag ของ Squid
Life_Miss (LM)	TCP_REFRESH_MISS
Life_Hit (LH)	TCP_REFRESH_HIT, TCP_IMS_HIT, TCP_IMS_MISS
Life_Unknown (LU)	TCP_HIT, TCP_MISS, TCP_MEM_HIT, TCP_CLIENT_REFRESH_MISS, UDP_HIT, UDP_MISS
Life_Ignore (LI)	NONE, ERR_*, TCP_DENIDE, TCP_NEGATIVE_HIT, TCP_SWAPFAIL

โดยที่

Life_Miss (LM) หมายถึง ผลของการร้องขอที่เกิดขึ้นหลังจากออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตมีการเปลี่ยนแปลงขนาดหรือวันที่ออบเจกต์นั้น ๆ ถูกสร้างขึ้นมา (Created Date)

Life_Hit (LH) หมายถึง ผลของการร้องขอที่เกิดขึ้นในขณะที่ออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตไม่มีการเปลี่ยนแปลงทั้งขนาดและวันที่ถูกสร้าง

Life_Unknown (LU) หมายถึง ผลของการร้องขอที่ยังไม่สามารถชี้ชัดได้ว่าออบเจกต์จริงที่อยู่ในอินเทอร์เน็ตมีการเปลี่ยนแปลงขนาดและ/หรือวันที่ถูกสร้างหรือไม่

Life_Ignore (LI) หมายถึง ผลของการร้องขอที่ไม่เกี่ยวข้องกับการเปลี่ยนแปลงของออบเจกต์ ซึ่งจะไม่นำมาประมวลผลแต่อย่างใด

ประเภทของการร้องขอที่จะนำมาใช้ในการคำนวณหาอายุของออบเจกต์นั้นได้แก่ประเภท Life_Miss, Life_Hit และ Life_Unknown ซึ่งในสองประเภทแรกสามารถนำไปคำนวณหาอายุได้เลยโดยจะจัดให้มีความแม่นยำของอายุที่หาได้จากกรณีนี้อยู่ในระดับที่ 1 (Accuracy Level 1) ทั้งนี้จะกำหนดให้ค่าอายุจริงของออบเจกต์มีความแม่นยำอยู่ในระดับ 0 (Accuracy Level 0) ส่วนในกรณีของผลของการร้องขอแบบ Life_Unknown นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะต้องมีการตรวจสอบก่อนว่าการร้องขอออบเจกต์ที่ทำให้เกิดกรณีของ Life_Unknown นั้นตัวออบเจกต์ที่ถูกร้องขอมีการเปลี่ยนแปลงขนาดหรือไม่ ถ้ามีการเปลี่ยนแปลงขนาดจึงจะถือว่าออบเจกต์นั้นมีการเปลี่ยนแปลงและจะอนุโลมให้การร้องขอนี้จัดอยู่ในประเภท Life_Miss เพื่อนำไปใช้ในการคำนวณต่อไปและจะจัดให้ความแม่นยำของค่าอายุที่หาได้จากกรณี Life_Unknown นี้อยู่ในระดับที่ 2 (Accuracy Level 2)

เราสามารถประมาณอายุของออบเจกต์ ในช่วงที่ i ($LifeCal_i$)ได้จากการนำเวลาของการเกิด Life_Miss ครั้งล่าสุด(LM_i) และ Life_Hit ครั้งล่าสุด(LH_i)มาหาค่าเฉลี่ยแล้วลบด้วยค่าเฉลี่ยของเวลาที่เกิดการเปลี่ยนแปลงของออบเจกต์ในช่วงก่อนหน้านั้น(LM_{i-1} , LH_{i-1}) ดังแสดงในสมการที่ 1

$$Lifecal_i = \frac{(LM_i + LH_i)}{2} - \frac{(LM_{i-1} + LH_{i-1})}{2} \qquad (1)$$

ในส่วนของขนาดของออบเจกต์นั้นเราสามารถหาได้จากฟิลด์ที่ 5 (Size) จากไฟล์ access.log โดยมีข้อแม้ว่าเราไม่ได้เก็บค่าขนาดของออบเจกต์ทุกๆ ค่าที่ปรากฏอยู่ในไฟล์นั้น แต่จะเก็บค่าขนาดของออบเจกต์ก็ต่อเมื่อพบว่าออบเจกต์นั้นๆ มีการเปลี่ยนแปลงเกิดขึ้นเท่านั้น (ในกรณีที่เกิดการร้องขอที่จัดอยู่ในประเภท Life_Miss) โดยจะนับจำนวนของอายุของ ออบเจกต์ไปด้วย

จากวิธีการคำนวณอายุของออบเจกต์ที่กล่าวมาข้างต้นสามารถสรุปออกมาเป็นอัลกอริธึมได้ดังนี้ ขั้นแรกเริ่มจากการจัดประเภทของการร้องขอออก ประเภทต่าง ๆ ตามผลของการร้องขอที่ปรากฏใน Log-Tag ของ Squid ซึ่งจะได้ผลออกมาเป็น 3 กรณีที่สามารถนำมาคำนวณหาอายุของออบเจกต์ดังนี้คือ

1.Life Hit Case (LH) ให้นำ URL ที่ถูกร้องขอนั้นไปหาในฐานข้อมูลว่ามี URL นั้นอยู่ในฐานข้อมูลหรือไม่ ถ้าไม่พบ แสดงว่าเป็น URL ใหม่ที่ไม่เคยมีการเก็บข้อมูลมาก่อน ให้ทำการเก็บบันทึกข้อมูล URL, เวลา ลงในฐานข้อมูล ถ้าพบว่ามี URL อยู่แล้วให้ทำการเปรียบเทียบว่าขนาดของ URL ปัจจุบันมีค่าเท่ากับขนาดของ URL ที่พบในฐานข้อมูลหรือไม่ ถ้าขนาดเท่ากัน ให้ทำการเพิ่มค่าจำนวนของการร้องขอที่มีเข้ามาในช่วงอายุนี้ (Count) และทำการเก็บค่าเวลาที่เกิดการร้องขอนี้ไว้เป็นเวลาล่าสุดที่มีการร้องขอออบเจกต์ เพื่อใช้ในการคำนวนเมื่อมีการร้องขออันถัดไปเข้ามา ถ้าพบว่าขนาดของ URL ที่พบไม่เท่ากับขนาดของ URL ปัจจุบันให้บันทึกลง Error log

2.Life Miss Case (LM) ให้นำ URL ที่ถูกร้องขอนั้นไปหาในฐานข้อมูลว่ามี URL นั้นอยู่ในฐานข้อมูลหรือไม่ ถ้าไม่พบแสดงว่าเป็น URL ใหม่ที่ไม่เคยมีการเก็บข้อมูลมาก่อน ให้ทำการเก็บบันทึกข้อมูล URL, เวลาที่เกิดการร้องขอนี้ซึ่งเป็นเวลาที่ถือว่าออบเจกต์นี้เกิดในช่วงอายุใหม่ (Birth) ลงในฐานข้อมูล ถ้าพบว่ามี URL อยู่แล้วให้ทำการเปรียบเทียบค่า เวลาที่ออบเจกต์เกิดในช่วงอายุล่าสุด

ถ้าเวลาที่ออบเจกต์เกิดในช่วงอายุล่าสุดมีค่าเท่ากับ 0 ซึ่งหมายความว่ายังไม่เคยมีการบันทึกการร้องขอที่ออบเจกต์มีการเกิดมาเลย ให้ทำการตรวจสอบเวลาในการร้องขอครั้งล่าสุดของออบเจกต์นั้นว่ามีค่าเท่ากับ 0 หรือไม่. ถ้าเวลาในการร้องขอครั้งล่าสุดของ ออบเจกต์นั้นเท่ากับ 0 (ไม่เคยมีการร้องขอออบเจกต์นี้มาก่อน) ให้กำหนดค่าเวลาในการเกิดของออบเจกต์นี้เท่ากับเวลาปัจจุบันที่มีการร้องขอนี้เข้า ถ้าเวลาในการร้องขอครั้งล่าสุดของออบเจกต์นั้นเท่าไม่กับ 0 ให้กำหนดค่าเวลาในการเกิดของออบเจกต์นี้จากค่าเฉลี่ยระหว่างเวลาปัจจุบันที่มีการร้องขอนี้กับเวลาในการร้องขอครั้งล่าสุดก่อนหน้าการร้องขอปัจจุบัน

จากนั้นให้ทำการกำหนดค่า การร้องขอครั้งล่าสุดของออบเจกต์นี้เท่ากับเวลาปัจจุบันและทำการบันทึกข้อมูลของ URL นี้และปิด Record ปัจจุบัน(จบช่วงอายุ) จากนั้นให้ทำการเปิด Record ใหม่ในฐานข้อมูลในฐานะของออบเจกต์เกิดใหม่(ช่วงอายุใหม่)โดยกำหนดให้ค่า เวลาในการเกิดของออบเจกต์ให้มีค่าเท่ากับที่คำนวณได้จากข้างต้นและกำหนดให้ค่าการร้องขอครั้งล่าสุดของออบเจกต์ในอายุใหม่นี้มีค่าเท่ากับเวลาปัจจุบันของการร้องขอออบเจกต์นี้ในช่วงอายุที่เพิ่งจบไป

ถ้าเวลาที่ออบเจกต์เกิดในช่วงอายุล่าสุดมีค่าไม่เท่ากับ 0 ซึ่งหมายความว่าออบเจกต์นี้เคยมีการเปลี่ยน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ห้ามนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แปลงเวลาเกิดมาแล้ว ให้ทำการกำหนดเวลาในการเกิดครั้งใหม่นี้โดยหาจากค่าเฉลี่ยระหว่างเวลาปัจจุบันที่มีการร้องขอนี้กับเวลาในการร้องขอครั้งล่าสุดก่อนหน้าการร้องขอปัจจุบัน เมื่อได้เวลาในการเกิดครั้งใหม่นี้แล้วก็สามารถนำไปคำนวณอายุของออบเจกต์ในช่วงนี้ได้จากการนำค่าเวลาในการเกิดที่เพิ่งจะคำนวณได้ในครั้งใหม่นี้ลบด้วยเวลาในการเกิดครั้งล่าสุด

จากนั้นให้ทำการกำหนดค่า การร้องขอครั้งล่าสุดของ ออบเจกต์นี้เท่ากับเวลาปัจจุบันและทำการบันทึกข้อมูล URL นี้และปิด Record ปัจจุบัน(จบช่วงอายุ) จากนั้นให้ทำการเปิด Record ใหม่ในฐานข้อมูลในฐานะของ ออบเจกต์เกิดใหม่ (ช่วงอายุใหม่) โดยกำหนดให้ค่า เวลาในการเกิดของออบเจกต์ให้มีค่าเท่ากับที่คำนวณได้จากข้างต้น และกำหนดให้ค่าการร้องขอครั้งล่าสุดของออบเจกต์ในอายุใหม่นี้มีค่าเท่ากับเวลาปัจจุบันของการร้องขอออบเจกต์นี้ในช่วงอายุที่เพิ่งจบไป

3.Life Unknown (LU) ให้นำ URL ที่ถูกร้องขอนั้นไปหาในฐานข้อมูลว่ามี URL นั้นอยู่ในฐานข้อมูลหรือไม่ ถ้าไม่พบ แสดงว่าเป็น URL ใหม่ที่ไม่เคยมีการเก็บข้อมูลมาก่อน ให้ทำการเก็บบันทึกข้อมูล URL, เวลา ลงในฐานข้อมูลและให้กำหนดระดับความแม่นยำเท่ากับ 2 (Accuracy Level =2) ถ้าพบว่ามี URL อยู่แล้วให้ทำการเปรียบเทียบว่าขนาดของ URL ปัจจุบันมีค่าเท่ากับขนาดของ URL ที่พบในฐานข้อมูลหรือไม่ ถ้าขนาดเท่ากัน ให้ไปอ่านข้อมูลบรรทัดต่อไปจาก Access.log และเริ่มต้นการทำงานที่จัดประเภทต่อไป ถ้าขนาดไม่เท่ากัน (มีการเปลี่ยนช่วงอายุเกิดขึ้น) ให้กำหนดระดับความแม่นยำเท่ากับ 2 (Accuracy Level =2) แล้วไปทำงานต่อในกรณีของ Life Miss Case (LM)

การหาค่าความผิดพลาดที่เกิดจากวิธีการหาอายุของออบเจกต์

ในงานวิจัยฉบับนี้จะมุ่งเสนอวิธีการหาความผิดพลาดในการคำนวณหาอายุจากการร้องขอที่มีความแม่นยำในระดับที่ 1 (Accuracy Level =1) เท่านั้น

จากรูปที่ 1 ข้างต้นเราสามารถคำนวณหาอายุของออบเจกต์ได้ตามสมการที่ 1 ซึ่งเป็นการคำนวณหาอายุจากค่าเฉลี่ย โดยที่เราไม่สามารถที่จะวัดค่าอายุจริง (Lifeactual) ของออบเจกต์นั้นได้โดยทราบแต่เพียงว่าจะมีค่าอยู่ระหว่างค่า ($LH_i - LM_{i-1}$) และ ($LM_i - LH_{i-1}$) นั้นคือ

$$\left(LH_i - LM_{i-1}\right) \le Lifeactual_i \le \left(LM_i - LH_{i-1}\right) \qquad (2)$$

จากสมการที่ 1 จะค่าอายุของออบเจกต์ในช่วงอายุที่ i จะได้ว่า

$$Lifecal_i = \frac{1}{2}\left[\left(LM_i - LH_{i-1}\right) + \left(LH_i - LM_{i-1}\right)\right] \qquad (3)$$

ดังนั้นเราสามารถหาค่าความผิดพลาดที่เกิดจากการหาอายุของออบเจกต์ที่ได้จากการคำนวณ (ΔL) จากการนำค่าอายุที่คำนวณได้ไปลบออกจากอายุจริงของออบเจกต์ นั้นคือ

$$\Delta L = Lifeactual_i - Lifecal_i \qquad (4)$$

นำสมการ (2) และ (3) มาแทนค่าในสมการ (4) จะได้

$$\left(\frac{(LH_i - LM_{i-1})}{2} - \frac{(LM_i - LH_{i-1})}{2}\right) \le \Delta L \le \left(\frac{(LM_i - LH_{i-1})}{2} - \frac{(LH_i - LM_{i-1})}{2}\right) \qquad (5)$$

พิจารณานิพจน์ทางซ้ายของอสมการที่ (5) จะได้

$$= -\frac{1}{2}\left[\left(LM_{i-1} - LH_{i-1}\right) + \left(LM_i - LH_i\right)\right] \qquad (5.1)$$

พิจารณานิพจน์ทางขวาของอสมการที่ (5) จะได้

$$= \frac{1}{2}\left[\left(LM_{i-1} - LH_{i-1}\right) + \left(LM_i - LH_i\right)\right] \qquad (5.2)$$

ดังนั้นจะได้ว่า

$$-a_i \le \Delta L \le a_i \qquad (6)$$

เมื่อ

$$a_i = \frac{1}{2}\left[\left(LM_{i-1} - LH_{i-1}\right) + \left(LM_i - LH_i\right)\right] \qquad (7)$$

จากสมการที่ (4) และ (6) จะได้ว่า

$$Lifecal_i = Lifeactual_i \pm a_i \qquad (8)$$

นั่นก็คือค่าอายุของออบเจกต์ที่คำนวณได้จะค่าเท่ากับอายุจริงของออบเจกต์บวกเข้ากับค่าความแม่นยำ (a_i)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวิเคราะห์ผลกระทบของอัตราการร้องขอข้อมูล (Request Rate) ที่มีผลกระทบต่อค่าความผิดพลาดของอายุที่ได้จากการคำนวณ

พิจารณาค่า a จากสมการที่ (7) จะพบว่าค่า a คือค่าเฉลี่ยของ ($LM_{i-1} - LH_{i-1}$) และ ($LM_i - LH_i$) ซึ่งแต่ละพจน์ก็คือช่วงห่างของแต่ละการร้องขอที่มีเข้ามาในขณะที่มีการเปลี่ยนช่วงอายุของออบเจกต์ ถ้าช่วงห่างของการร้องขอมากจะทำให้โอกาสที่จะพบการเปลี่ยนช่วงอายุของออบเจกต์มีน้อยลงซึ่งผลที่ตามมาคือค่าอายุที่วัดได้จะมีโอกาสผิดพลาดมากขึ้น ดังนั้นเราสามารถหาค่า a ในเชิงสถิติได้จากอัตราส่วนของจำนวนการร้องขอที่มีเข้ามาในแต่ละช่วงอายุต่ออายุจริงของออบเจกต์จาก

$$a_i = \frac{Lifeactual_i}{N + 1} \quad (9)$$

โดยที่

N คือจำนวนของการร้องขอประเภท LH ที่มีเข้ามาในช่วงระหว่าง LM_{i-1} ถึง LM_i และ ค่าN_i ต้องมีค่ามากกว่า 0 ($N > 0$) ทั้งนี้เนื่องจากรูปที่ 1 จะเห็นว่าถ้าค่า N_iเป็น 0 ก็จะทำให้ LH_i หายไปทำให้ไม่สามารถคำนวณหาค่าอายุของออบเจกต์ได้อย่างถูกต้องแม่นยำ

แทนค่า a จากสมการที่ (9) ลงในสมการที่ (8) จะได้ว่า

$$Lifecal_i = Lifeactual_i \pm \left(\frac{Lifeactual_i}{N_i + 1}\right)$$

$$Lifecal_i = Lifeactual_i\left(1 \pm \frac{1}{N_i + 1}\right) \quad (10)$$

จากสมการที่ (9) จะเห็นว่าถ้าออบเจกต์นั้นมีการถูกร้องขอ (N_i) มากขึ้นก็จะทำให้ a มีค่าน้อยลงจนเข้าใกล้ 0 และจะทำให้อายุที่คำนวณได้มีค่าแม่นยำมากขึ้นตามสมการที่ 10 นั่นคือ

$$\lim_{N_i \to \infty} Lifecal_i = Lifeactual_i \quad (11)$$

จากวิธีการคำนวณหาอายุของออบเจกต์ด้วยวิธีดังกล่าวนั้นเราสามารถที่จะหาความแม่นยำของการคำนวณอายุแต่ละออบเจกต์ออกมาเป็นค่าเปอร์เซ็นต์ของความแม่นยำ (%accuracy) ได้โดยการกำหนดค่าอายุที่เราทราบแน่นอนและทดลองสร้างการร้องขอออบเจกต์นั้นเพื่อที่จะหาอัตราส่วนของค่า a_i ต่อค่าอายุจริง(ที่เรากำหนด)

$$\%acc = \pm\left(\frac{a_i}{Lifeactual_i}\right)x100\% \quad (12)$$

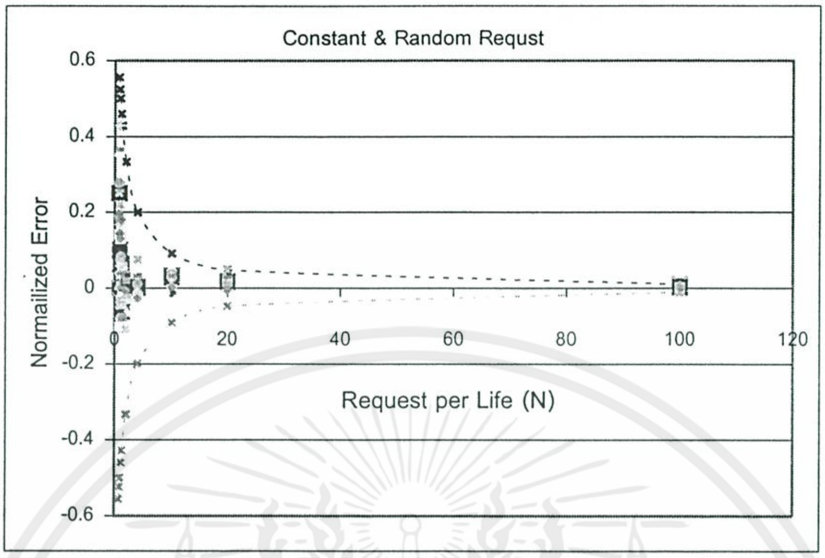
แทนค่า a จากสมการที่ (9) จะได้ว่า

$$\%acc = \pm\left(\frac{1}{N_i + 1}\right)x100\% \quad (13)$$

ผลการทดลองหาค่าความผิดพลาดที่เกิดจากวิธีการหาอายุของออบเจกต์

การทดลองที่ 1 ทำการทดลองสร้างการร้องขอที่มีอัตราคงที่ต่อออบเจกต์ที่มีอายุคงที่เป็นจำนวน 3 ช่วง โดยแปรค่าอายุตั้งแต่ 30 ถึง 300 วินาที (เนื่องจากต้องการประหยัดเวลาในการทดลองจึงเลือกกำหนดช่วงอายุของออบเจกต์ที่มีค่าน้อย ๆ) จำนวน 10 ค่า และแปรจำนวนการร้องขอ 0.8 ถึง 100 ครั้งต่อ 1 ช่วงอายุจำนวน 10 ค่า โดยนำค่าอายุที่คำนวณได้ 3 ค่าจากในแต่ละช่วงอายุมาทำการเฉลี่ยก่อนนำไปหาผลต่างจากช่วงอายุจริง และทำการทดลองข้างต้นซ้ำเหมือนเดิมอีก 1 ชุดแต่จะสร้างการร้องขอแบบสุ่มที่มีค่าเฉลี่ยของอัตราการร้องขอตั้งแต่ 0.8 ครั้ง ถึง 100 ครั้งต่อ 1 ช่วงอายุจำนวน 10 ค่า

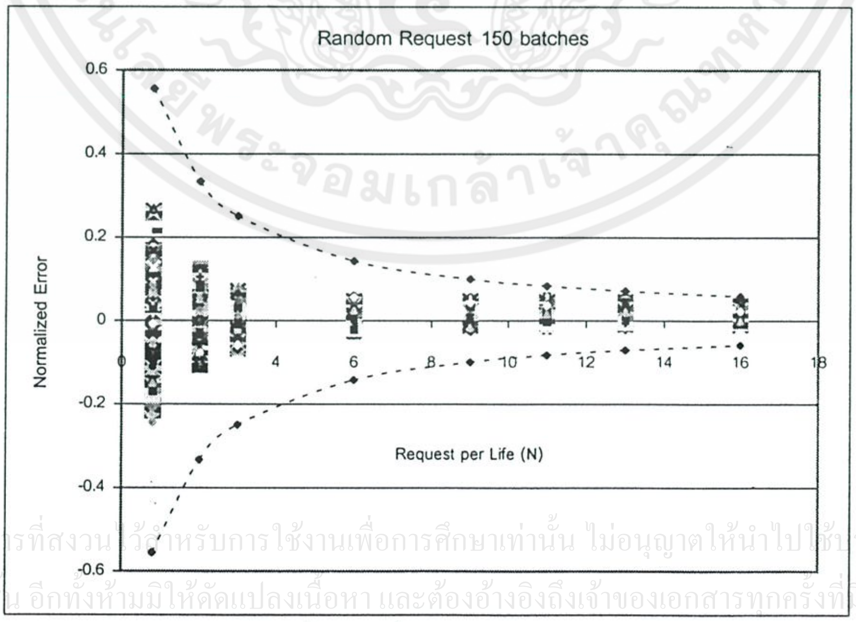
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 แสดงค่าความผิดพลาดของอายุของออบเจกต์ซึ่งถูกนอร์มอลไลซ์ (Normalized) ณ.อัตราการร้องขอต่าง ๆ

จากรูปกราฟจะเห็นได้ว่าเมื่อจำนวนครั้งในการร้องขอต่อ 1 ช่วงอายุของออบเจกต์เพิ่มขึ้น ค่าความผิดพลาดที่เกิดขึ้นก็จะมีช่วงของความผิดพลาดเข้าใกล้ 0 มากขึ้นหรือมีความผิดพลาดน้อยลง นั้นหมายถึงว่ายิ่งมีจำนวนการร้องขอต่อ 1 ช่วงอายุของออบเจกต์มากขึ้นค่าอายุที่คำนวณที่ได้ก็จะมีความแม่นยำมากขึ้น และจะสังเกตค่าความผิดพลาดที่ปรากฏจากกราฟว่ามีค่าอยู่ในกรอบของค่า a ซึ่งแสดงด้วยเส้นประในรูปกราฟ

การทดลองที่ 2 ทำการทดลองการหาค่าความผิดพลาดในการหาอายุของออบเจกต์จากการคำนวณโดยทำการทดลองซ้ำในแต่ละช่วงอายุตั้งแต่ 5, 10, 20, 50, 100, 120 และ 150 วินาที เป็นจำนวนช่วงอายุละ 150 ครั้ง โดยมีการแปรอัตราการร้องขอแบบสุ่มที่มีค่าเฉลี่ยอยู่ที่ 0.8, 2, 3, 6, 9, 11, 13 และ 16 ครั้งต่อ 1 ช่วงอายุแล้วทำการหาความผิดพลาดของอายุที่เกิดขึ้นมาทำการนอร์มอลไลซ์



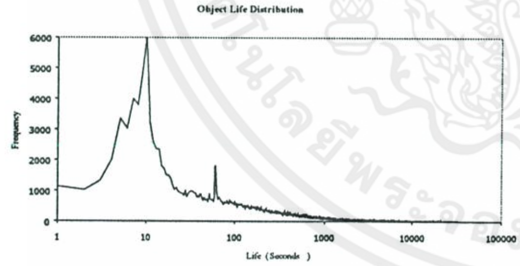
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3 แสดงค่าความผิดพลาดของอายุที่คำนวณซึ่งถูกนอร์มอลไลซ์ ณ.อัตราการร้องขอใน 1 ช่วงอายุต่าง ๆ

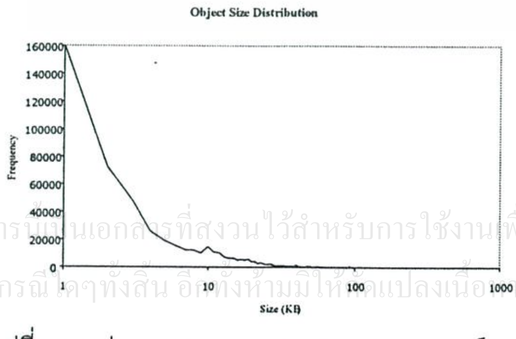
จากผลการทดลองที่แสดงในกราฟจากรูปที่ 2 และ 3 จะเห็นได้ว่าอายุของออบเจกต์ที่วัดได้จะมีค่าความผิดพลาดอยู่ในช่วงที่ระบุไว้คือไม่เกินค่า a และ ถ้าอัตราการร้องขอมีความถี่มากขึ้นจะทำให้ค่าความผิดพลาดของค่าอายุที่วัดได้ลดลง

ผลการทดลองหาอายุและขนาดของออบเจกต์จาก Log File ของพร็อกซีเซิร์ฟเวอร์ที่ใช้งานจริง

จากวิธีการหาอายุและขนาดของออบเจกต์ข้างต้นได้นำมาพัฒนาโปรแกรมโดยใช้ Perl Script [4] เพื่อช่วยในการประมวลผล ซึ่งในการทดลองนั้นจะใช้ Perl Script ที่ได้มาประมวลผล Log File จากเครื่อง proxy1.kmitl.ac.th ซึ่งทำหน้าที่เป็นพร็อกซีเซิร์ฟเวอร์หลักของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง ในช่วงเวลาตั้งแต่วันที่ 1 ถึง 14 กุมภาพันธ์ พ.ศ.2543 ซึ่งคิดเป็นข้อมูลการร้องขอ 9,422,162 requests (เฉลี่ยวันละ 673,011 requests) มาทำการหาการกระจายของขนาดและอายุของออบเจกต์โดยได้ผลดังแสดงในรูปที่ 4 และ 5



รูปที่ 4 กราฟแสดงการกระจายของขนาดของออบเจกต์



รูปที่ 5 กราฟแสดงการกระจายของอายุของออบเจกต์

ผลการทดลองแสดงให้เห็นถึงการกระจายตัวของขนาดของออบเจกต์ส่วนใหญ่จะมีค่าไม่เกิน 10 กิโลไบต์ โดยมีค่าเฉลี่ยอยู่ที่ 8.78 กิโลไบต์ ในส่วนการกระจายอายุของออบเจกต์ส่วนมากมีค่าน้อยกว่า 10 วินาทีโดยมีค่าเฉลี่ยอยู่ที่ 19128.29 วินาที (ประมาณ 5 ชั่วโมง 19 นาที) [5] โดยมีความผิดพลาดเฉลี่ยทุกออบเจกต์ที่มี Accuracy Level 1 อยู่ที่ ±26.69% ทั้งนี้มี ออบเจกต์จำนวน 347,440 ออบเจกต์ที่มีขนาดอยู่ระหว่าง 0 ถึง 1 กิโลไบต์ และอีก 885 ออบเจกต์ที่มีอายุอยู่ระหว่าง 0 ถึง 1 วินาที เนื่องจากในความเป็นจริงมีออบเจกต์บางตัวที่มีขนาดไม่ถึง 1 กิโลไบต์ (เช่น มีขนาด 250 ไบต์) และ/หรือ มีอายุไม่ถึง 1 วินาที

อนึ่งการคำประมวลผลจาก Log File ดังกล่าวทำงานบนเครื่อง HP LH4 Dual Pentium III Xeon 500MHz ใช้เวลาในการประมวลผลรวม 172 ชั่วโมง

สรุป

เราสามารถวัดอายุของออบเจกต์ได้จาก Log file ของ Squid ตามอัลกอริธึมในหัวข้อวิธีการหาอายุและขนาดของออบเจกต์และระบุความแม่นยำของการวัดได้ตามสมการ (12) โดยที่ค่าความผิดพลาดที่ได้จากการวัดอายุของออบเจกต์จะมีค่าลดลงเมื่อมีอัตราการร้องขอออบเจกต์ในช่วงอายุนั้น ๆ มากขึ้น

นอกจากนี้แล้วยังสามารถนำวิธีการวัดค่าอายุของออบเจกต์นี้ไปใช้กับการวัดค่าจาก Log fileประเภทอื่น ๆ ที่มีเวลากำกับและสามารถจัดประเภทของแต่ละ record ของ Log file ได้ว่ามีความเกี่ยวข้องกับค่าที่ต้องการจะวัดหรือไม่ รวมทั้งสามารถหาค่าความผิดพลาดของค่าที่วัดได้อีกด้วย

เอกสารอ้างอิง

[1] Cormack, Andrew, 1996. Web Caching,[Online] Available, URL:http://www.jisc.ac.uk/acn/caching.html

[2] Wessels, D., 2000, Squid Cache Server, [Online] Available, URL: http://www.nlanr.net/Squid/index.html

[3] Wessels, D., 2000, Squid Frequently Ask Questions, [Online] Available, URL:http://www.squid-cache.org/Doc/FAQ/FAQ-6.html#ss6.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[4] Schwartz, Radal L., 1994, Unix Programming: Learning Perl, O'reilly & Associates, Inc.

[5] วิฑิต อินทรหะ, อัครินทร์ คุณกิตติ, "การวิเคราะห์ลักษณะของวัตถุใน World Wide Web จาก Squid Log", First National Symposium on Graduate Research Chiangmai University, Jun 2000.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน นายวิฑิต อินทรหะ

วันเดือนปีเกิด 25 มีนาคม 2516

สถานที่เกิด กรุงเทพมหานคร

ประวัติการศึกษา

ศึกษาระดับมัธยมปลาย โรงเรียน บดินทรเดชา (สิงห์ สิงหเสนีย์) ปีการศึกษา 2532

ศึกษาระดับปริญญาตรี วิทยาศาสตร์บัณฑิต สาขาเทคโนโลยีอาหาร/ เทคโนโลยีชีวภาพ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2536

ผลงานทางวิชาการที่ได้รับการตีพิมพ์

1 การวิเคราะห์ลักษณะของวัตถุใน World Wide Web จาก Squid Log

2 การวัดขนาดและอายุของออบเจกต์ใน World Wide Web จาก Log File

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้