

สถาปัตยกรรมเว็บแคชชนิดกำหนดบทบาทหน้าที่

An Architecture for Role Based Web Caching

วุฒิพล พงศ์พัฒนกิจโชติ¹ และสุรศักดิ์ สงวนพงษ์¹

บทคัดย่อ

ประสิทธิภาพของแคชเซิร์ฟเวอร์ไม่เพียงขึ้นกับขนาดของแคชหรือเครือข่ายแต่ขึ้นกับประเภทผู้ใช้งานด้วย (Marshall และ Roadknight, 1998) แคชเซิร์ฟเวอร์แบบดั้งเดิมให้บริการผู้ใช้โดยไม่จัดลำดับความสำคัญและไม่แบ่งแยกหน้าที่ ขณะที่ในทางปฏิบัติจริงมักพบว่าผู้ใช้ในองค์กรหนึ่งๆ มีบทบาทหน้าที่และระดับความสำคัญต่างกัน แคชเซิร์ฟเวอร์ดั้งเดิมจึงไม่ตอบสนองกับการใช้งานจริงในบางองค์กร

งานวิจัยนี้เสนอสถาปัตยกรรมของแคชเซิร์ฟเวอร์ชนิดกำหนดบทบาทหน้าที่ โดยการวิเคราะห์และจัดแบ่งผู้ใช้ให้มี สิทธิ หน้าที่ และระดับความสำคัญที่แตกต่างกัน รวมทั้งเสนอแนะขั้นตอนวิธีการทดแทนข้อมูลแคชที่ได้ดัดแปลงให้เหมาะสมกับสถาปัตยกรรม (Priority GreedyDual-Size) ผู้วิจัยได้พัฒนาแคชเซิร์ฟเวอร์ต้นแบบและวัดประสิทธิภาพการทำงานและพบว่าแคชเซิร์ฟเวอร์ชนิดกำหนดบทบาทหน้าที่ผนวกกับขั้นตอนวิธีการทดแทนข้อมูลช่วยให้ฮีต เรโซของกลุ่มผู้ใช้ที่มีระดับความสำคัญสูงมีค่าเพิ่มขึ้นจากปกติถึง 4.72% โดยไม่มีผลกระทบต่อแคชโดยรวม

คำสำคัญ แคชเซิร์ฟเวอร์ สิทธิ หน้าที่ ระดับความสำคัญ นโยบายทดแทนข้อมูล

ABSTRACT

Performance of the cache servers does not only rely on cache size or network, but also appears to be related to the different types of users (Marshall and Roadknight, 1998). However, traditional cache servers still handle all users with the same level of priority and profile.

This research proposed the architectural design of the Role Based Cache Server and priority cache replacement algorithm, by defining different types of rights, roles, and priorities for users. Researcher developed the prototype of cache server based on this design, and measured cache performance. Role Based Cache Server and priority cache replacement algorithm increased hit ratio of high priority users up to 4.72%, with a negligible impact to total cache performance.

Key Words right, role, priority, cache server, replacement policy

¹ กลุ่มงานวิจัยเครือข่ายประยุกต์ ภาควิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเกษตรศาสตร์

คำนำ

แคชเซิร์ฟเวอร์เป็นวิธีหนึ่งที่นิยมใช้อย่างแพร่หลายเพื่อลดความล่าช้าในการเข้าถึงข้อมูลและประหยัดวงจรรีเสิร์ฟขององค์กร (Alan Flavel, 1997) เมื่อผู้ใช้ร้องขอข้อมูล แคชเซิร์ฟเวอร์จะตรวจสอบว่ามีข้อมูลเก็บอยู่ในระบบหรือไม่ หากตรวจพบก็จะนำส่งข้อมูลให้ผู้ร้องขอโดยตรงซึ่งช่วยให้ไม่ต้องสูญเสียการเข้าชมของสัญญาณของวงจรรีเสิร์ฟ ในกรณีที่แคชเซิร์ฟเวอร์ไม่ได้บรรจุข้อมูลที่ผู้ใช้ร้องขอ แคชเซิร์ฟเวอร์จะร้องขอข้อมูลจากเซิร์ฟเวอร์ปลายทางเพื่อเก็บเข้าสู่ระบบและนำส่งไปยังผู้ใช้

จากผลการวิจัย (Kroeger, 1997) โดยการวิเคราะห์ค่าอัตราเร็วจากการจำลองการร้องขอเว็บไซต์ที่พีจำนวน 24,659,182 ครั้งจากผู้ใช้จำนวน 17,354 ราย ซึ่งร้องขอไปยังเซิร์ฟเวอร์ปลายทางจำนวน 140,506 เครื่อง พบว่าการใช้งานแคชจะช่วยลดความล่าช้าได้ 22% ถึง 26% และค่าอัตราเร็วก็จะมีได้สูงสุดเพียง 47% ถึง 52%

(Marshall และ Roadknight, 1998) และ (Ghaleb Abdulla, Edward Fox และ Marc Abrams, 1997) ศึกษาการร้องขอเว็บไซต์ที่พีจากกลุ่มผู้ใช้ต่างๆ เพื่อพิจารณาความเหมือนและแตกต่างของรูปแบบการร้องขอข้อมูล พบว่าประสิทธิภาพของแคชเซิร์ฟเวอร์จะแตกต่างกันไปแต่ละเซิร์ฟเวอร์ และโดยส่วนใหญ่พบว่าประสิทธิภาพของแคชจะไม่เพียงขึ้นกับความจุของระบบเก็บข้อมูลเพียงอย่างเดียว หากแต่ยังขึ้นกับกลุ่มของผู้ใช้งาน อีกทั้งการร้องขอส่วนมาก

มักเกิดจากผู้ใช้น้อยที่ติดต่อไปยังเซิร์ฟเวอร์ปลายทางจำนวนเล็กน้อยเท่านั้น

แนวทางการปรับปรุงประสิทธิภาพของแคชเซิร์ฟเวอร์ในอีกมุมหนึ่งได้แก่การศึกษาถึงขั้นตอนวิธีการทดแทนข้อมูลแคช (Cache Replacement Policy) งานการวิจัยของ (Cao และ Irani, 1997) และ (Junho Shim, Peter Scheuermann และ Radek Vingralek, 1999) ได้เปรียบเทียบขั้นตอนวิธีการทดแทนข้อมูลแคชจำนวน 10 ขั้นตอนวิธี และได้เสนอขั้นตอนวิธี GreedyDual-Size ซึ่งปรับปรุงมาจากขั้นตอนวิธี GreedyDual (Young, 1991) โดยใช้ทั้งขนาดของข้อมูลและระยะเวลาการดาวน์โหลดในการพิจารณาเลือกทดแทนข้อมูล ผลการทดลองพบว่า GreedyDual-Size มีประสิทธิภาพสูงกว่าขั้นตอนวิธีการทดแทนแคชแบบอื่น (Cao และ Irani, 1997) โดยมีอัตราเร็วที่สูงกว่า และทำงานได้ดีมากในระบบที่มีแคชขนาดเล็กอีกด้วย

อย่างไรก็ตามแนวทางการพัฒนาประสิทธิภาพแคชทั้งสองด้านยังคงใช้หลักการพื้นฐานประการหนึ่งคือผู้ใช้งานแคชไม่มีความแตกต่างใดๆ ทั้งในแง่ประเภทของการขอบริการสิทธิ หรือระดับความสำคัญ รูปแบบนี้ย่อมมีผลในทางปฏิบัติกับองค์กรที่มีช่องสัญญาณสื่อสารจำกัด เพราะผู้ใช้งานที่มีความจำเป็นยิ่งยวดต้องอยู่ในแถวคอยเท่าเทียมกับผู้ใช้งานที่มีความจำเป็นน้อยหรือไม่มีความจำเป็นที่ต้องใช้งาน การออกแบบแคชเซิร์ฟเวอร์ในงานวิจัยนี้มีจุดประสงค์เพื่อยกระดับประสิทธิภาพการขอบริการแคชสำหรับผู้ที่มีระดับความสำคัญสูงโดยการบูร

ถนอมการแนวความคิดของ สิทธิ หน้าที่ และระดับ ความสำคัญไว้ในการให้บริการและการทำงาน ของแคช งานวิจัยนี้มีส่วนร่วมในการพัฒนา ประสิทธิภาพของแคชเซิร์ฟเวอร์ในอีกมิติหนึ่ง ผู้วิจัยได้ออกแบบโครงสร้างสถาปัตยกรรมของ แคชเซิร์ฟเวอร์ชนิดกำหนดบทบาทหน้าที่และ ออกแบบขั้นตอนวิธีการทดแทนข้อมูลแคชที่ พิจารณาถึงความสำคัญของผู้ใช้ และพัฒนา ต้นแบบของโปรแกรมแคชเซิร์ฟเวอร์ และวิเคราะห์ ผลการทดลองจากข้อมูลแคชเซิร์ฟเวอร์ที่ใช้ใน สภาพแวดล้อมจริง

วัตถุประสงค์ และวิธีการ

วิธีการที่ใช้ประกอบด้วยสามส่วน คือ หัวข้อที่ 1 การออกแบบสถาปัตยกรรมระบบ หัวข้อที่ 2 การออกแบบขั้นตอนวิธีทดแทนข้อมูล และหัวข้อที่ 3 การพัฒนาโปรแกรมแคชเซิร์ฟเวอร์ ต้นแบบและการวิเคราะห์ผล

1. การออกแบบสถาปัตยกรรมระบบ

สถาปัตยกรรมของเว็บแคชชนิดกำหนด บทบาทหน้าที่ออกเป็นส่วนได้ดังนี้

1.1. การติดต่อระหว่างไคลเอนต์และแคช เซิร์ฟเวอร์

หัวข้อนี้กล่าวถึงวิธีการสื่อสารที่ไคลเอนต์ จะขอใช้บริการจากแคชเซิร์ฟเวอร์ซึ่งแสดงได้ดัง รูปที่ 1 ขั้นตอนการติดต่อระหว่างไคลเอนต์และ แคชเซิร์ฟเวอร์ออกเป็น 3 ส่วน ดังนี้

1.1.1 การพิสูจน์ทราบผู้ใช้ (Authentication)

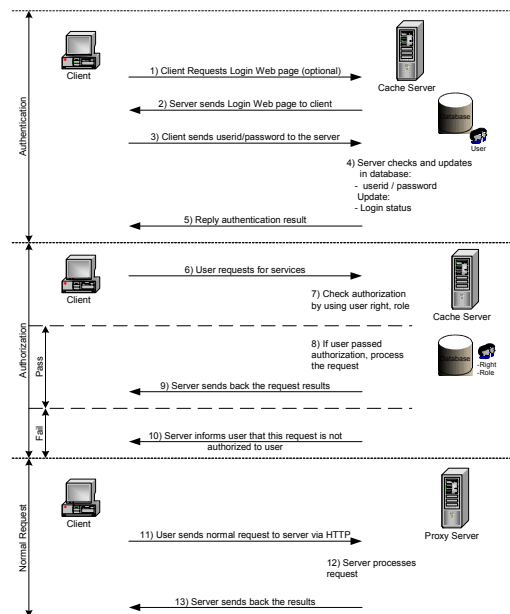
เป็นการพิสูจน์ทราบบุคคลที่ขอใช้งาน แคชเซิร์ฟเวอร์ ผู้ใช้ทุกคนจำเป็นต้องผ่านขั้นตอน นี้และสามารถผ่านไปขั้นตอนต่อไปเมื่อแคชเซิร์ฟเวอร์ อนุญาตเท่านั้น

1.1.2 การตรวจสอบสิทธิผู้ใช้ (Authorization)

เป็นการตรวจสอบว่าผู้ใช้มีสิทธิหรือ หน้าที่ในการร้องขอบริการนั้นๆ หรือไม่ โดยผู้ใช้ จะผ่านขั้นตอนนี้ได้เมื่อผู้ใช้มีสิทธิหรือหน้าที่ที่ สามารถร้องขอบริการนั้นๆ รูปแบบของบริการจะ กล่าวในหัวข้อ ที่ 2.2

1.1.3 การร้องขอบริการ (Normal Request)

ขั้นตอนนี้ที่ผู้ใช้สามารถร้องขอบริการ ทั่วไปจากแคชเซิร์ฟเวอร์ เช่นการร้องขอเว็บเพ จปกติที่รวมการให้บริการแคช เป็นต้น



รูปที่ 1 แผนภาพการติดต่อระหว่าง ไคลเอนต์และเซิร์ฟเวอร์

1.2. ประเภทของบริการ

ตารางที่ 1 แสดงถึงตัวอย่างต้นแบบการจัดประเภทของบริการที่แคชเซิร์ฟเวอร์ชนิดกำหนดบทบาทหน้าที่ ผู้ดูแลระบบแคชสามารถปรับแต่งหรือแก้ไขหรือเพิ่มเติมประเภทและรายละเอียดการให้บริการ ได้ตามความเหมาะสมของแต่ละองค์กร

ตารางที่ 1 ประเภทบริการต่างๆ ของแคชเซิร์ฟเวอร์ชนิดกำหนดบทบาทหน้าที่

No	ประเภทบริการ	รายละเอียด
บริการจัดการการร้องขอของผู้ใช้		
1	Submit Cache Request	การให้บริการร้องขอการบริการแคช
2	Change User Cache Request Queue	การให้บริการเปลี่ยนแปลงแถวการร้องขอบริการแคชของผู้ใช้เอง
3	Change Group Cache Request Queue	การให้บริการเปลี่ยนแปลงแถวการร้องขอบริการแคชของกลุ่ม
บริการจัดการข้อมูลผู้ใช้		
1	Add User	การให้บริการเพิ่มผู้ใช้
2	Remove User	การให้บริการลบผู้ใช้
3	Modify User	การให้บริการเปลี่ยนแปลงข้อมูลผู้ใช้
บริการจัดการข้อมูลกลุ่มผู้ใช้		
1	Add Group	การให้บริการเพิ่มกลุ่มผู้ใช้
2	Remove Group	การให้บริการลบกลุ่มผู้ใช้
3	Modify Group	การให้บริการเปลี่ยนแปลงข้อมูลกลุ่มผู้ใช้
บริการจัดการการทำงานแคช		
1	View User Cache Database	การให้บริการดูฐานข้อมูลแคชของผู้ใช้
2	View Group Cache Database	การให้บริการดูฐานข้อมูลแคชของกลุ่มผู้ใช้
3	Force User Cache Expiration	การให้บริการกำหนดให้ข้อมูลในแคชของผู้ใช้หมดอายุ
4	Force Group Cache Expiration	การให้บริการกำหนดให้ข้อมูลในแคชของกลุ่มผู้ใช้หมดอายุ
5	Force User Cache Retention	การให้บริการกำหนดให้รักษาข้อมูลในแคชของผู้ใช้
6	Force Group Cache Retention	การให้บริการกำหนดให้รักษาข้อมูลในแคชของกลุ่มผู้ใช้
7	Refresh User Cache Item	การให้บริการปรับปรุงสถานะข้อมูลในแคชของผู้ใช้
8	Refresh Group Cache Item	การให้บริการปรับปรุงสถานะข้อมูลในแคชของกลุ่มผู้ใช้
บริการจัดการตัวเลือกการทำงานแคช		
1	Set User Embedded Cache Status	การให้บริการกำหนดการแสดงผลสถานะการแคชของผู้ใช้
2	Set Group Embedded Cache Status	การให้บริการกำหนดการแสดงผลสถานะการแคชของกลุ่มผู้ใช้

1.3. สิทธิ หน้าที่ และระดับความสำคัญ

แคชเซิร์ฟเวอร์สามารถให้บริการที่แตกต่างกันต่อผู้ใช้งานแต่ละรายดังนี้

1.3.1 สิทธิ

สิทธิคือบริการพิเศษที่ผู้ใช้หรือกลุ่มของผู้ใช้สามารถร้องขอจากแคชเซิร์ฟเวอร์ สิทธิของผู้ใช้จะปรับให้อยู่ในแนวเดียวกับบริการที่แคชเซิร์ฟเวอร์มีให้เพื่อความสะดวกในการจำแนกสิทธิในการร้องขอบริการ ดังนั้นสิทธิของผู้ใช้จึงจำแนกได้ดังตารางที่ 1

1.3.2 หน้าที่

หน้าที่คือกลุ่มของสิทธิที่ประกอบกันเพื่อแสดงถึงบทบาทต่างๆ ที่ผู้ใช้หรือกลุ่มของผู้ใช้มีต่อแคชเซิร์ฟเวอร์ หน้าที่ที่ประกอบด้วยสิทธิมากกว่าหนึ่งอย่างที่ประกอบกันเป็นบทบาทต่างๆ ข้อดีของการประกอบสิทธิต่างๆ ให้เป็นหน้าที่คือความสะดวกในการจัดการกับกลุ่มของสิทธิที่อยู่ในหมวดเดียวกัน ตัวอย่างของหน้าที่แสดงไว้ในตารางที่ 2

1.3.3 ระดับความสำคัญ

ระดับความสำคัญคือค่าความสำคัญที่แคชเซิร์ฟเวอร์จะให้บริการต่อผู้ใช้ ผู้วิจัยกำหนดระดับความสำคัญต้นแบบจากความสำคัญมากที่สุดไปน้อยที่สุดเป็น 5 ระดับ โดยค่าระดับความสำคัญจะนำไปใช้ในขั้นตอนวิธีเลือกเปลี่ยนข้อมูลแคชต่อไป

ตารางที่ 2 หน้าที่ของผู้ใช้และกลุ่มผู้ใช้

No	หน้าที่	สิทธิ
1	Userid Administrator	Add User Remove User Modify User
2	Group Administrator	Add Group Remove Group Modify Group
3	Cache User	Submit Cache Request Set User Embedded Cache Status
4	Group Cache User	Set Group Embedded Cache Status
5	User Cache Administrator	View User Cache Database Force User Cache Expiration Force User Cache Retention Refresh User Cache Item Change User Cache Request Queue
6	Group Cache Administrator	View Group Cache Database Force Group Cache Expiration Force Group Cache Retention Refresh Group Cache Item Change Group Cache Request Queue

1.4. ผู้ใช้และกลุ่มของผู้ใช้

ผู้ใช้และกลุ่มของผู้ใช้สามารถจัดแบ่งได้ดังนี้

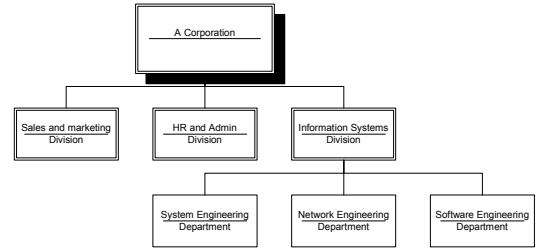
1.4.1 ผู้ใช้

แนวความคิดของผู้ใช้เปรียบได้กับบุคลากรในองค์กร ผู้ใช้จะต้องสังกัดอยู่ในกลุ่มผู้ใช้อย่างน้อยหนึ่งกลุ่มซึ่งเปรียบได้กับบุคลากรในองค์กรที่ต้องสังกัดแผนกต่างๆ และจะมีสิทธิหน้าที่ และระดับความสำคัญแตกต่างกันในแต่ละกลุ่มที่ผู้ใช้สังกัด

1.4.2 กลุ่มผู้ใช้

กลุ่มผู้ใช้มีเพื่ออำนวยความสะดวกในการบริหารข้อมูลและการทำงานของผู้ใช้ที่มีบทบาทสัมพันธ์กัน ลักษณะนี้เปรียบได้กับแผนกต่างๆ ในองค์กร กลุ่มผู้ใช้สามารถนำมาจัดเป็นลำดับชั้นดังรูปที่ 2 และกลุ่มผู้ใช้แต่ละกลุ่มมี

คุณสมบัติพื้นฐานเช่น ชื่อกลุ่ม ระดับความสำคัญของกลุ่มในองค์กร เป็นต้น



รูปที่ 2 โครงสร้างกลุ่มผู้ใช้ที่เปรียบได้กับองค์กร

2. การออกแบบขั้นตอนวิธีทดแทนข้อมูล

ผู้วิจัยเสนอขั้นตอนวิธี Priority GreedyDual-Size ซึ่งเป็นขั้นตอนวิธีที่ปรับปรุงมาจาก GreedyDual-Size (Cao และ Irani, 1997) โดยพิจารณาเพิ่มระดับความสำคัญของข้อมูลในแคชซึ่งเชื่อมโยงกับผู้ใช้แต่ละราย ขั้นตอนวิธีนี้จะพิจารณาถึงค่าใช้จ่าย H ที่มีค่าเท่ากับค่าใช้จ่ายในการดาวน์โหลดข้อมูลหารด้วยขนาดข้อมูล

เมื่อแคชเต็มและระบบจำเป็นต้องเลือกข้อมูลเพื่อที่จะลบออกและแทนที่ด้วยข้อมูลใหม่ ข้อมูลที่มีค่า H น้อยที่สุดหรือค่า $H(q)$ จะถูกแทนที่ด้วยข้อมูลใหม่ และข้อมูลอื่นๆ ทั้งหมดในแคชจะถูกลดค่า H ลงเป็น $H(q)$ แต่ถ้าข้อมูล p ได้รับอ้างอิง ระบบจะปรับค่า $H(p)$ ให้กลับเป็นค่าใช้จ่ายเดิมที่เกิดขึ้นในการดาวน์โหลดข้อมูล p มาสู่ในแคช การทำงานรูปแบบนี้จะลดค่า H ของข้อมูลทุกๆ ตัวลงเมื่อเวลาผ่านไป และจะคืนค่า H ให้กับข้อมูลที่ได้รับการอ้างอิง ทำให้ขั้นตอนวิธีนี้พิจารณาครอบคลุมทั้ง Locality และ

ค่าใช้จ่ายที่ใช้ อย่างมีประสิทธิภาพ การคำนวณค่า H ที่พิจารณาถึงค่าระดับความสำคัญจะเป็นดังนี้

$$H(p) = L + [c(p) / s(p)] * P_p$$

โดยที่ P_p เป็นตัวแปรระดับความสำคัญของข้อมูล p และ $0 < P_p$ โดยที่ค่า P_p ของผู้ใช้งานแต่ละรายสามารถคำนวณได้จากระดับความสำคัญของผู้ใช้

ถ้า P_p มีค่ามาก จะทำให้ $H(p)$ มีค่าสูงกว่าปกติ แต่ถ้า P_p มีค่าน้อยจะทำให้ $H(p)$ มีค่าน้อยกว่าปกติ ทำให้การพิจารณาค่า $H(p)$ เพื่อเลือกข้อมูลที่จะลบออกจากแคช ได้มีการปรับค่าให้ตรงกับระดับความสำคัญของข้อมูล ตัวอย่างของขั้นตอนวิธี Priority GreedyDual-Size จะแสดงไว้ในรูปที่ 3

```

Algorithm Priority GreedyDual-Size:
Initialize L = 0
Process each request document in turn:
The current request is for document p:
If p is already in memory,
    H(p) = L + [c(p) / s(p)] * P_p
If p is not in memory,
    While there is not enough room in
    memory for p,
        Let L = min_q H(q)
        Evict q such that H(q) = L
        Bring p into memory and set
            H(p) = L + [ c(p) / s(p) ] * P_p
End
    
```

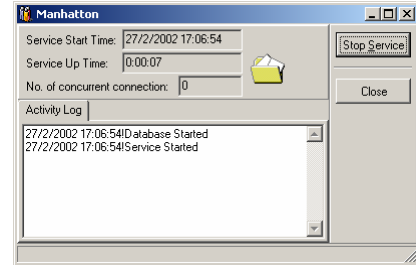
รูปที่ 3 ขั้นตอนวิธี Priority GreedyDual-Size

3. การพัฒนาโปรแกรมแคชเซิร์ฟเวอร์ต้นแบบและการวิเคราะห์

3.1. แคชเซิร์ฟเวอร์ต้นแบบ

ต้นแบบของแคชเซิร์ฟเวอร์ชนิดกำหนดบทบาทหน้าที่ซึ่งพัฒนาขึ้นโดยใช้ C++ และใช้ชื่อ Manhattan โดยใช้สถาปัตยกรรมที่กล่าวไว้ข้างต้น และผนวกขั้นตอนวิธี Priority

GreedyDual-Size รูปที่ 4 แสดงหน้าจอการทำงาน

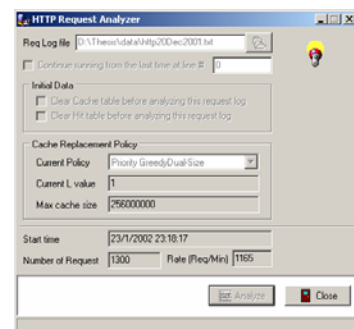


รูปที่ 4 โปรแกรม Manhattan

3.2. โปรแกรมวิเคราะห์ผล

โปรแกรม HTTP Request Analyzer ดังแสดงรูปที่ 5 เป็นโปรแกรมที่ใช้ในการวิเคราะห์ประสิทธิภาพแคช โดยมีรูปแบบในการวัดค่าดังนี้

- 1) วัดค่าอัตราเร็วของแคชเมื่อใช้ขั้นตอนการเลือกแทนที่ข้อมูลในแคชเมื่อแคชเต็ม โดยเปรียบเทียบระหว่างขั้นตอนวิธี GreedyDual-Size และ Priority GreedyDual-Size เพื่อวิเคราะห์ผลที่มีต่อประสิทธิภาพของแคชโดยรวม
- 2) วัดค่าอัตราเร็วของกลุ่มผู้ใช้ที่มีความสำคัญต่างกันเพื่อเปรียบเทียบค่าการเปลี่ยนแปลงของอัตราเร็วระหว่างกลุ่มผู้ใช้ที่มีความสำคัญสูงและกลุ่มผู้ใช้ที่มีความสำคัญต่ำ เพื่อวิเคราะห์ผลกระทบที่มีต่อประสิทธิภาพของแคชของกลุ่มผู้ใช้ที่มีความสำคัญต่างกัน



รูปที่ 5 โปรแกรม HTTP Request Analyzer

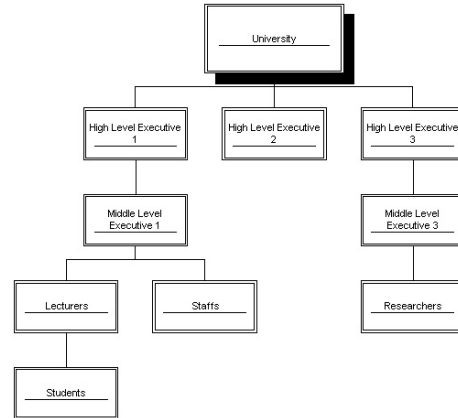
ข้อมูลที่ใช้ในการทดสอบเลือกเก็บจากข้อมูลการร้องขอจากเซิร์ฟเวอร์ของมหาวิทยาลัยเกษตรศาสตร์ ตารางที่ 3 แสดงพารามิเตอร์ที่สำคัญของข้อมูลการร้องขอเอชทีทีพีที่ใช้ในการวิเคราะห์ประสิทธิภาพแคช

การร้องขอเอชทีทีพีจากพร็อกซีเซิร์ฟเวอร์จากตารางที่ 3 อยู่ในคาบเวลาหนึ่งสัปดาห์ (วันที่ 19 ธันวาคม ค.ศ.2001 ถึงวันอังคารที่ 25 ธันวาคม ค.ศ.2001) ซึ่งครอบคลุมการใช้งานทั้งเวลาว่างปกติและวันหยุด คาบเวลาหนึ่งสัปดาห์ที่เลือกใช้สามารถเป็นตัวแทนข้อมูลโดยรวมได้เนื่องจากได้วิเคราะห์แล้วพบว่าข้อมูลการร้องขอเอชทีทีพีมีแบบแผนเดียวกันในทุกสัปดาห์

ตารางที่ 3 ข้อมูลการร้องขอเอชทีทีพี

พารามิเตอร์	ค่า
จำนวนการร้องขอข้อมูลเอชทีทีพี	178,670
ระยะเวลาในการร้องขอ	19/12/2001 01:45:24 ถึง 25/12/2001 02:14:44
ขนาดของแคช	256 MB
จำนวนของผู้ใช้	ประมาณ 1,000

รูปที่ 6 แสดงโครงสร้างของกลุ่มผู้ใช้แคชที่จำลองขึ้นเทียบกับโครงสร้างของมหาวิทยาลัย การจำแนกและแบ่งหมวดหมู่เพื่อกำหนดกลุ่มผู้ใช้ตามบทบาทหน้าที่ใช้วิธีจัดแบ่งตามไอพีแอดเดรสของไคลเอนต์ โดยกำหนดให้แต่ละไอพีแอดเดรสแทนผู้ใช้หนึ่งราย และให้กลุ่มเครือข่ายย่อยแทนกลุ่มผู้ใช้ ผู้วิจัยกำหนดลำดับหรือระดับความสำคัญของกลุ่มผู้ใช้แต่ละกลุ่มให้แตกต่างกันตามโครงสร้างขององค์กรและจำนวนผู้ใช้งานเพื่อกำหนดกลุ่มผู้ใช้ที่มีความสำคัญสูง และกลุ่มผู้ใช้ที่มีความสำคัญต่ำเพื่อใช้ในการเปรียบเทียบประสิทธิภาพการทำงานของแคชดังตารางที่ 4



รูปที่ 6 โครงสร้างองค์กรของกลุ่มผู้ทดลอง

ตารางที่ 4 กลุ่มผู้ใช้งานในการทดลอง

กลุ่มที่	ชื่อกลุ่ม	ลำดับ	จำนวนผู้ใช้	ไอพีแอดเดรส
1	High Level Executive 1	80	295	158.108.9...
2	High Level Executive 2	80	136	158.108.6...
3	High Level Executive 3	80	269	158.108.7...
4	Middle Level Executive 1	65	362	158.108.4...
5	Middle Level Executive 3	65	345	158.108.5...
6	Lecturers	50	639	158.108.2...
7	Staffs	35	949	158.108.8...
8	Researcher	60	499	158.108.3...
9	Students	20	1154	158.108.1...

ผลการทดลองและวิจารณ์

ผลการจำลองการร้องขอเอชทีทีพีและวัดค่าฮิตเรโซของแคชพบว่าค่าฮิตเรโซของแคชโดยรวมจากการใช้งานขั้นตอนวิธี GreedyDual-Size มีค่าเท่ากับ 34.98% และค่าฮิตเรโซของแคชโดยรวมจากการใช้งานขั้นตอนวิธี Priority GreedyDual-Size เท่ากับ 34.13% ดังตารางที่ 5 การใช้งานขั้นตอนวิธี Priority GreedyDual-Size ทำให้ประสิทธิภาพของแคชโดยรวมลดลงเพียง 0.85% เท่านั้น ซึ่งถือได้ว่าน้อยมากจนไม่มีผลกระทบต่อประสิทธิภาพแคชโดยรวม

ตารางที่ 5 ผลการทดลองค่าฮิตเรโซของแคช

ขั้นตอนวิธี	Hit	Miss	Total	Hit Ratio
GreedyDual-Size	62501	116169	178670	34.98125
Priority GreedyDual-Size	60980	117690	178670	34.12996

การวัดประสิทธิภาพการทำงานของขั้นตอนวิธี Priority GreedyDual-Size เมื่อนำไปใช้เพิ่มประสิทธิภาพของกลุ่มผู้ใช้ที่มีความสำคัญทำโดยการวัดค่าฮิตเรโซเปรียบเทียบกับระหว่างการใช้งานขั้นตอนวิธี GreedyDual-Size และ Priority GreedyDual-Size ของกลุ่มผู้ใช้ที่มีความสำคัญต่างๆ แสดงผลการทดลองได้ในตารางที่ 6

ในที่นี้พบว่าเมื่อใช้ขั้นตอนวิธี Priority GreedyDual-Size ค่าฮิตเรโซของผู้ใช้ที่มีความสำคัญสูงมีค่าเพิ่มขึ้น เช่นกลุ่มผู้ใช้ Middle Level Executive 1 ที่ค่าฮิตเรโซเพิ่มขึ้นจาก 32.22% เป็น 36.95% หรือเพิ่มขึ้น 4.72% หรือกลุ่มผู้ใช้ High Level Executive 1 ที่ค่าฮิตเรโซเพิ่มขึ้นจาก 28.17% เป็น 31.96% หรือเพิ่มขึ้น 3.79% ถือได้ว่าสูงอย่างมีนัยสำคัญเมื่อเทียบกับค่าฮิตเรโซปกติซึ่งมีค่าประมาณ 30% ถึง 40%

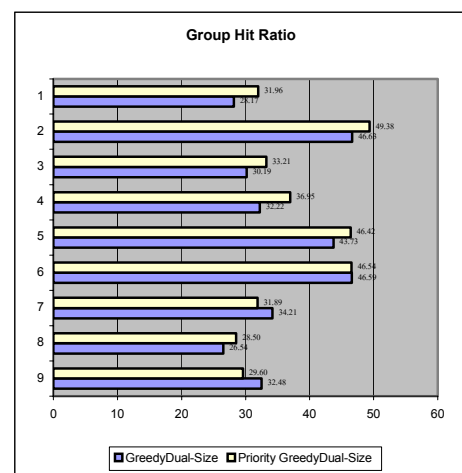
ถึงแม้ว่าค่าฮิตเรโซของกลุ่มผู้ใช้ที่มีความสำคัญต่ำจะมีค่าลดลงบ้างเช่นกลุ่ม Staffs ที่มีค่าฮิตเรโซลดลงจาก 34.21% ไปเป็น 31.89% หรือลดลง 2.32% หรือกลุ่มผู้ใช้ Students ที่ลดลงจาก 32.48% เป็น 29.60% หรือลดลง 2.88% แต่โดยรวมแล้วประสิทธิภาพของแคชโดยรวมยังคงไม่เปลี่ยนแปลง

ตารางที่ 6 ผลการวัดค่าฮิตเรโซของกลุ่มผู้ใช้

กลุ่ม	ชื่อกลุ่ม	ลำดับ	จำนวนผู้ใช้	Hit Ratio
-------	-----------	-------	-------------	-----------

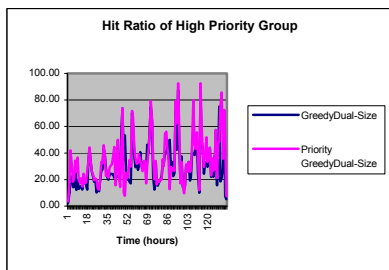
	กลุ่ม		Greedy Dual-Size	Priority GreedyDual-Size
1	High Level Executive 1	80	28.17	31.96
2	High Level Executive 2	80	46.63	49.38
3	High Level Executive 3	80	30.19	33.21
4	Middle Level Executive 1	65	32.22	36.95
5	Middle Level Executive 3	65	43.73	46.42
6	Lecturers	50	46.59	46.54
7	Staffs	35	34.21	31.89
8	Researcher	60	26.54	28.50
9	Students	20	32.48	29.60

จากข้อมูลในตารางที่ 6 สามารถนำมาเปรียบเทียบค่าฮิตเรโซของกลุ่มผู้ใช้เมื่อใช้งานทั้งสองขั้นตอนวิธี โดยจะเห็นได้ว่ากลุ่มผู้ใช้ที่มีระดับความสำคัญสูง (กลุ่มที่ 1-5 และ 8) จะมีฮิตเรโซจากการใช้ขั้นตอนวิธี Priority GreedyDual-Size สูงกว่าเมื่อใช้ขั้นตอนวิธี GreedyDual-Size กลุ่มผู้ใช้ที่มีระดับความสำคัญปานกลาง (กลุ่มที่ 6) จะมีฮิตเรโซเกือบจะเท่ากันระหว่างการใช้สองขั้นตอนวิธี และกลุ่มผู้ใช้ที่มีระดับความสำคัญต่ำ (กลุ่มที่ 7 และ 9) จะมีฮิตเรโซของการใช้ขั้นตอนวิธี Priority GreedyDual-Size ต่ำกว่าการใช้ขั้นตอนวิธี GreedyDual-Size ดังแสดงในรูปที่ 7

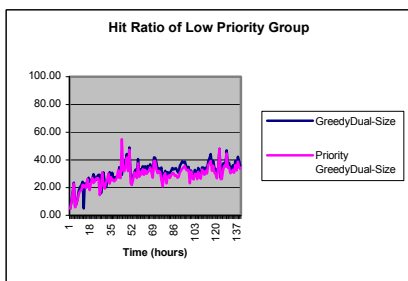


รูปที่ 7 เปรียบเทียบฮิตเรโซของกลุ่มผู้ใช้

การเปรียบเทียบค่าอัตราเร็วระหว่างการใช้งานขั้นตอนวิธี GreedyDual-Size และ Priority GreedyDual-Size เทียบกับระยะเวลาของกลุ่มผู้ใช้ที่มีความสำคัญสูงแสดงได้ดังรูปที่ 8 พบว่าเกือบทุกคาบเวลาอัตราเร็วของการใช้ ขั้นตอนวิธี Priority GreedyDual-Size จะมีค่าสูงกว่าอัตราเร็วของ GreedyDual-Size ซึ่งสอดคล้องกับผลลัพธ์ที่ได้จากตารางที่ 6 ในทางตรงข้ามการเปรียบเทียบค่าอัตราเร็วเทียบกับระยะเวลาของกลุ่มผู้ใช้ที่มีความสำคัญต่ำดังรูปที่ 9 แสดงให้เห็นว่าในเกือบทุกคาบเวลาอัตราเร็วของการใช้ ขั้นตอนวิธี Priority GreedyDual-Size จะมีค่าต่ำกว่าอัตราเร็วของการใช้ขั้นตอนวิธี GreedyDual-Size และเป็นการยืนยันผลการทดลองที่ได้ในตารางที่ 6



รูปที่ 8 กราฟเปรียบเทียบค่าอัตราเร็วเทียบกับระยะเวลาของกลุ่มผู้ใช้ที่มีความสำคัญสูง



รูปที่ 9 กราฟเปรียบเทียบค่าอัตราเร็วเทียบกับระยะเวลาของกลุ่มผู้ใช้ที่มีความสำคัญต่ำ

จากผลการทดลองสามารถสรุปได้ว่า ขั้นตอนวิธี Priority GreedyDual-Size สามารถช่วยเพิ่มประสิทธิภาพแคชให้กลุ่มผู้ใช้ที่มีความสำคัญสูงโดยที่ไม่มีผลต่อประสิทธิภาพแคชโดยรวมและแคชเซิร์ฟเวอร์ที่พิจารณาถึงบทบาทหน้าที่และแคชขั้นตอนวิธีที่คำนึงถึงระดับความสำคัญของผู้ใช้สามารถทำงานได้อย่างยืดหยุ่นและสามารถปรับใช้กับสภาพแวดล้อมที่แท้จริงขององค์กรต่างๆ ที่ต้องการให้บริการแคชที่แตกต่างกันสำหรับผู้ใช้ ผู้พัฒนาสามารถนำกรอบการทำงานนี้ไปออกแบบและพัฒนาเพิ่มเติมเพื่อเพิ่มประสิทธิภาพและความยืดหยุ่นให้ตรงกับการใช้งานจริงในองค์กรต่างๆ ได้มากที่สุดต่อไป

ในด้านการพัฒนาในอนาคตมีแนวทางในการพัฒนาออกเป็นสองด้าน ดังนี้

1. การพัฒนาต่อเนื่องในแนวตั้ง ซึ่งเป็นการเพิ่มประสิทธิภาพและความสามารถโดยการเพิ่มบริการแก่ผู้ใช้งาน การเพิ่มความสะดวกในการบริหารเซิร์ฟเวอร์ หรือการปรับปรุงขั้นตอนวิธีที่ใช้ในการแคช

2. การพัฒนาต่อเนื่องในแนวนอน เป็นการเพิ่มประสิทธิภาพโดยการพัฒนาให้สนับสนุนการทำงานแบบคลัสเตอร์ การทำกระจายการทำงานหรือการพัฒนาระบบสำรองการทำงาน เป็นต้น

เอกสารอ้างอิง

Alan J. Flavell, *WWW Cache Briefing*, Department of Physics and Astronomy, Glasgow University, 1997.

Ian Marshall and Chris Roadknight, *Linking Cache Performance to User Behaviour*. In Proceedings of the 3rd International WWW Caching Workshop, Manchester, England, 1998.

Ghaleb Abdulla, Edward A. Fox, Marc Abrams, *Shared User Behavior on the*

World Wide Web, WebNet97, Toronto, Canada, October 1997.

N. Young, *The k -server dual and lose competitiveness for paging*. *Algorithmica*, June 1994, vol. 11,(no.6):525-41. Rewritten version of "Online caching as cache size varies", in The 2nd Annual ACM-SIAM Symposium on Discrete Algorithms, 241-250,1991