หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 เว็บแคช

เว็บแคชจะทำหน้าที่เก็บสำเนาเอกสารที่เคยมีการร้องขอ เพื่อที่จะหลีกเลี่ยงการถ่ายโอนข้อมูลเอกสารเหล่านั้นอีกครั้งเมื่อมีการร้องขอในภายหลัง เว็บแคชจึงช่วยลดปริมาณการถ่ายโอนข้อมูลผ่านระบบเครือข่ายอินเทอร์เน็ต และทำให้เวลาที่ใช้ดึงเอกสารลดลงด้วย เว็บแคชแบ่งออกเป็น 3 กลุ่ม คือ ระบบแคชของเว็บบราวเซอร์ ระบบเว็บแคชของพร็อกซีเซิร์ฟเวอร์ และระบบแคชของเว็บเซิร์ฟเวอร์ โดยแคชแต่ละแบบรองรับปริมาณการร้องขอที่แตกต่างกันออกไป และมีเป้าหมายต่างกัน แต่มีหลักการทำงานที่เหมือนกัน

พร็อกซีแคชสามารถทำได้ทั้งในลักษณะเปิดเผย (Explicit) และลักษณะโปร่งใส (Transparent) แบบแรกผู้ใช้จำเป็นต้องตั้งค่าในบราวเซอร์เพื่อที่จะส่งการร้องขอไปยังพร็อกซีแบบที่สองผู้ใช้ไม่จำเป็นต้องตั้งค่าใดๆ เลย โดยอุปกรณ์ในระบบเครือข่ายเมื่อมีการตรวจพบการร้องขอไปยังทีซีพีพอร์ต 80 ก็จะทำการส่งต่อการร้องขอไปยังแคช เนื่องจากเอกสารที่เก็บอยู่ในแคชอาจจะเก่า เพราะว่าเมื่อมีการเปลี่ยนแปลงเอกสารต้นฉบับที่เซิร์ฟเวอร์ เซิร์ฟเวอร์ไม่ได้แจ้งให้แคชทราบ ดังนั้นการตัดสินใจว่าเอกสารนั้นยังใหม่อยู่หรือไม่ พิจารณาได้จากเวลาที่เอกสารเปลี่ยนแปลงครั้งล่าสุด (Last modification time) และเวลาที่มีการดึงเอกสารครั้งล่าสุด (The time of last retrieval) หรือการตรวจสอบว่าเอกสารยังใช้ได้หรือไม่ (Validation) ถ้าเอกสารยังใหม่อยู่แคชจะส่งเอกสารกลับให้ผู้เรียกขอได้ทันที แต่ถ้าเอกสารนั้นเก่าแล้ว แคชจะทำการตรวจสอบกับเซิร์ฟเวอร์ที่เก็บเอกสารต้นฉบับ โดยผลที่ได้จะเป็นสำเนาของเอกสารปัจจุบัน หรือรหัสที่บอกว่าเอกสารไม่มีการเปลี่ยนแปลง นอกจากนี้พร็อกซีแคชยังสามารถทำการติดต่อสื่อสารเพื่อแลกเปลี่ยนข้อมูลระหว่างกันได้อีกด้วย เพื่อทำให้มีโอกาสที่จะพบเอกสารมากขึ้น โดยมีรูปแบบการติดต่อสื่อสารหลายแบบ แต่ละแบบก็มีข้อดีข้อเสีย แตกต่างกัน

2.1.2 ความสอดคล้องกันของแคช (Cache Coherency) [11]

เนื่องจากมีเอกสารเป็นจำนวนมากที่เก็บอยู่ในเว็บแคช เอกสารเหล่านี้อาจถูกเก็บมาเป็นเวลานานแล้ว เมื่อมีผู้ใช้งานเรียกขอเอกสารดังกล่าวผู้ใช้งานอาจจะได้รับเอกสารที่ล้าสมัยไปแล้ว ซึ่ง ณ เวลานั้นเอกสารต้นฉบับที่เก็บอยู่ที่เว็บเซิร์ฟเวอร์อาจจะมีการเปลี่ยนแปลงแก้ไขเกิดขึ้น ดัง

นั้น เว็บแคชทุกๆ ตัว จึงควรมีกลไกในการทำให้เอกสารเหล่านี้ทันสมัยอยู่เสมอ ซึ่งจะทำให้ผู้ใช้งานได้รับเอกสารที่ใหม่เท่าที่จะเป็นไปได้ ก่อนจะกล่าวถึงกลไกต่างๆ จะขอกล่าวถึงคำสั่งของ HTTP [5] ที่ช่วยให้เว็บพร็อกซีมีความสอดคล้องกันคือ

- HTTP GET จะทำการดึงเอกสารตามยูอาร์เอลที่ได้รับ
- Condition GET คือ HTTP GET ที่มี IF-Modified-Since เป็นส่วนหัว โดยพร็อกซีจะใช้ date ในการบอกเซิร์ฟเวอร์ให้ส่งสำเนาเอกสารเฉพาะที่มีการเปลี่ยนแปลง
- Pragma:no-cache เป็นส่วนหัวที่ได้เพิ่มเข้าไปใน GET เพื่อกำหนดให้ดึงเอกสารอีกครั้งจากเซิร์ฟเวอร์ไม่ว่าเอกสารนั้นจะมีการเปลี่ยนแปลงหรือไม่
- Last-Modified:date จะถูกส่งกลับมาพร้อมกับทุกๆ การ GET โดยจะบอกเวลาล่าสุดที่เอกสารมีการเปลี่ยนแปลง
- Date:date เป็นเวลาล่าสุดที่เอกสารถูกพิจารณาว่ายังใหม่อยู่

สำหรับกลไกในการทำให้เกิดความสอดคล้องกันของแคชในปัจจุบันมี 2 รูปแบบ คือ Strong Cache Consistency และ Weak Cache Consistency

1) Strong Cache Consistency แบ่งออกเป็น

   Client Validation วิธีการนี้เรียกอีกอย่างหนึ่งว่า polling-every-time โดย พร็อกซีจะทำให้เกิด out-of-date ในการเข้าถึงข้อมูลแต่ละครั้ง และจะทำการส่งส่วนหัว If-Modified-Since ในการเข้าถึงข้อมูลแต่ละครั้งด้วย ซึ่งวิธีการนี้ทำให้เกิดการตอบกลับมาของเซิร์ฟเวอร์ว่า 304 (เป็นรหัสการตอบกลับของ HTTP สำหรับ "Not Modified") ถ้าข้อมูลไม่มีการเปลี่ยนแปลง

   Server Invalidation เมื่อมีการตรวจพบว่าข้อมูลมีการเปลี่ยนแปลงเซิร์ฟเวอร์จะทำการส่งข้อความบอกไคลแอนท์ทุกๆ เครื่อง ที่เพิ่งเข้าถึงข้อมูลว่า ข้อมูลที่ได้ไปมีการเปลี่ยนแปลงขึ้นแล้ว วิธีการนี้เซิร์ฟเวอร์จะต้องคอยตรวจสอบว่ามีไคลแอนท์เครื่องใดบ้างที่ใช้ข้อมูลที่มีการเปลี่ยนแปลงขึ้นแล้ว ซึ่งเป็นผลให้เซิร์ฟเวอร์ทำงานได้ช้าลงเมื่อมีจำนวนไคลแอนท์มาก

2) Weak Cache Consistency แบ่งออกเป็น

   Adaptive TTL ใช้จัดการกับปัญหาโดยการปรับ time-to-live ของเอกสาร ซึ่งดูจากช่วงชีวิตของเอกสาร เอกสารที่ไม่มีการเปลี่ยนแปลงมาเป็นเวลานานถือว่ามีโอกาสสูงที่เอกสารยังไม่เปลี่ยนแปลง ดังนั้น time-to-live ของเอกสารจึงถูกกำหนดเป็นเปอร์เซ็นต์ของอายุของเอกสารปัจจุบัน คือ เปอร์เซ็นต์ x (เวลาปัจจุบัน – เวลาที่ครั้งล่าสุดที่มีการเปลี่ยนแปลงของเอกสาร)

Piggyback Invalidation มีวิธีการ 3 แบบ คือ Piggyback Cache Validation (PCV) เมื่อมีการร้องขอจากพร็อกซีไปยังเซิร์ฟเวอร์ พร็อกซีจะส่งรายละเอียดของแคชติดไปด้วย เพื่อที่จะตรวจสอบเอกสารต่างๆ ภายในแคช อีกวิธีคือ Piggyback Server Invalidation (PSI) ใช้สำหรับเซิร์ฟเวอร์เพื่อส่งรายละเอียดว่ามีเอกสารใดเปลี่ยนแปลงตั้งแต่การเข้าถึงข้อมูลครั้งสุดท้ายของไคลแอนท์ติดกลับมาตอบพร็อกซี แล้วพร็อกซีจะตรวจสอบ และสามารถเพิ่มช่วงชีวิตของเอกสารที่ไม่ได้อยู่ในรายละเอียดได้ วิธีสุดท้ายก็แบบผสมระหว่างสองแบบแรก ซึ่งขึ้นอยู่กับเวลาที่พร็อกซีเรียกขอการตรวจสอบครั้งสุดท้าย ถ้าเวลาน้อยก็จะใช้ PSI ถ้ามีช่องว่างของเวลามากก็จะใช้ PCV

2.1.3 สถาปัตยกรรมของการแคช (Caching Architectures) [11]

ประสิทธิภาพของเว็บแคชขึ้นอยู่กับจำนวนไคลแอนท์ที่เชื่อมต่อกับเว็บแคช ยิ่งมีผู้ใช้งานมากก็จะมีโอกาสมากที่จะพบเอกสารที่ถูกเรียกขอก่อนหน้านี้ การทำงานร่วมกันของแคชหลายๆ เครื่อง จะทำให้โอกาสในการพบเอกสารมากขึ้นด้วย สถาปัตยกรรมของการแคชจึงช่วยให้การทำงานของพร็อกซีแคชแต่ละตัวมีประสิทธิภาพยิ่งขึ้น ซึ่งสามารถแบ่งสถาปัตยกรรมของการแคชออกเป็น 3 ประเภทคือ

1) แบบลำดับชั้น วิธีการหนึ่งที่ทำให้แคชทำงานร่วมกันได้คือ การแบ่งแคชออกเป็นหลายๆ ระดับ ซึ่งสมมุติให้แคชมี 4 ระดับ คือ Bottom, Institutional, Regional และ National ณ ระดับ Bottom จะมีไคลแอนท์แคช เมื่อการร้องขอไม่พบเอกสารในไคลแอนท์แคชก็จะส่งผ่านไปให้ ระดับ Institutional ถ้าไม่พบอีกก็จะส่งผ่านไปเรื่อยๆ จนถึงระดับ National ถ้ายังไม่พบเอกสารที่ถูกร้องขอในทุกๆ ระดับ แคชระดับ National ก็จะติดต่อไปยังเซิร์ฟเวอร์ที่เก็บเอกสารโดยตรง เมื่อค้นพบเอกสารแล้วก็จะถูกส่งต่อกลับลงมาตามระดับ และทำสำเนาเอกสารไว้ในทุกๆ ระดับด้วย
2) แบบกระจาย ในระบบเว็บแคชแบบกระจายจะไม่มีแคชระหว่างกลาง จะมีเพียงแคชระดับ Institutional เท่านั้นซึ่งจะคอยจัดการเมื่อไม่พบเอกสารในแคชแต่ละตัว แคชระดับ institutional จะเก็บข้อมูลต่างๆ เกี่ยวกับเนื้อหาที่มีอยู่ในแคชทุกๆ ตัวที่ทำงานร่วมกันไว้ เพื่อที่จะใช้ตัดสินว่าแคชระดับ Institutional ตัวไหนที่จะต้องไปดึงเอกสารที่ค้นไม่พบ แคชแบบกระจายนี้ยังทำให้เกิดการแชร์โหลดที่ดีกว่า และมีการรองรับความผิดพลาดได้มากกว่าด้วย โดยวิธีการติดต่อสื่อสารกัน

ระหว่างแคชมีอยู่หลายวิธีด้วยกัน เช่น ICP [6], CARP [7], CRISP [8], Cachemesh [9] เป็นต้น

3) แบบผสม จะมีการทำงานร่วมกันของแคชทั้งในระดับเดียวกัน และคนละระดับ โดยใช้ไอซีพีในการติดต่อกันระหว่างแคช โดยเอกสารที่จะถูกดึงต้องเป็นเอกสารที่ใช้เวลาในการเดินทางน้อยที่สุด และในบางกรณีการดึงเอกสารจากเซิร์ฟเวอร์โดยตรงเลยก็อาจจะมีค่าใช้จ่ายที่ต่ำกว่าด้วย

2.1.4 ไอซีพี (ICP - Internet Cache Protocol) [6]

ไอซีพีเป็นข้อความชนิด Lightweight message ใช้สำหรับติดต่อสื่อสารกันระหว่างเว็บแคช โดยแคชจะทำการแลกเปลี่ยน ICP query และ ICP reply เพื่อรวบรวมข้อมูลในการใช้เลือกว่าที่ที่เหมาะสมที่จะดึงข้อมูลตั้งอยู่ที่ใด หลักการทำงานคือ เมื่อแคชได้รับการร้องขอจากผู้ใช้แล้วก็จะทำการตรวจสอบว่ามีเอกสารที่ต้องการเก็บไว้ในแคชหรือไม่ ถ้ามีก็จะส่งให้ผู้เรียกขอ ถ้าไม่มีก็จะส่งข้อความไปถามแคชข้างเคียงทุกตัวในระบบ แล้วรอผลตอบกลับมาถ้าหากพบว่ามีเอกสารอยู่ภายในแคชตัวอื่นก็จะทำการร้องขอไปยังแคชตัวนั้น ถ้าหากไม่พบก็จะส่งข้อความไปถามแคชในระดับที่สูงกว่าต่อไป ถ้าหากไม่พบเอกสารในแคชใดๆ เลยก็จะดึงเอกสารจากเซิร์ฟเวอร์โดยตรง

2.1.5 ขั้นตอนวิธีการแทนที่ (Replacement Algorithm) [11]

ในแคชทุกๆ แบบจะมีขั้นตอนวิธีการแทนที่เพื่อใช้ในการพิจารณาเลือกแทนที่เอกสารในแคชเมื่อมีเอกสารใหม่เข้ามาในขณะที่แคชมีที่ว่างไม่เพียงพอ เพื่อให้เอกสารภายในแคชทันสมัยและมีโอกาสสูงที่จะถูกเรียกขอในอนาคต ขั้นตอนวิธีการแทนที่ที่นำมาใช้ในงานวิจัยมีดังนี้

- FIFO (First In First Out) แทนที่เอกสารที่ถูกเก็บตามลำดับก่อนหลัง
- LRU (Least Recently Used) แทนที่เอกสารที่ไม่ถูกเรียกใช้มานานที่สุดก่อน
- SIZE แทนที่เอกสารที่มีขนาดใหญ่ที่สุดก่อน

2.1.6 การทดสอบเปรียบเทียบประสิทธิภาพของเว็บแคช

ในการเปรียบเทียบประสิทธิภาพการจำลองการทำงานของเว็บแคชจะใช้ค่าฮิตเรท และไบต์ฮิตเรทเป็นเครื่องวัด ซึ่งคำนวณได้จากสูตร

$$\text{ฮิตเรท} = \frac{\text{จำนวนการร้องขอข้อมูลซึ่งพบในแคช}}{\text{จำนวนการร้องขอข้อมูลทั้งหมด}}$$

$$\text{ไบต์ฮิตเรท} = \frac{\text{จำนวนไบต์ของการร้องขอข้อมูลซึ่งพบในแคช}}{\text{จำนวนไบต์ของการร้องขอข้อมูลทั้งหมด}}$$

2.1.7 เอ็นเอสทู (NS-2 : Network Simulator version 2) [3]

เอ็นเอสทูเป็นซอฟต์แวร์ที่ใช้ในโปรเจควินท์ (VINT) เพื่อใช้จำลองการทำงานของระบบเครือข่าย โดยเป้าหมายของโปรเจควินท์ คือเพื่อที่จะทำให้เกิดความง่ายในการวิจัยทางด้านระบบเครือข่าย และมุ่งเน้นระบบเครือข่ายที่ขยายได้ และการปฏิสัมพันธ์กันระหว่างหลายๆ โพรโทคอล แกนหลักของเอ็นเอสทูพัฒนาขึ้นมาจากภาษาซีพลัสพลัส จึงมีลักษณะเป็นส่วนประกอบต่างๆ ซึ่งส่วนประกอบที่เอ็นเอสทูสนับสนุนมีอย่างเช่น โพรโทคอล โทโพโลจี โหนด ลิงค์ และเว็บแคช เป็นต้น โดยทั้งหมดใช้ภาษาโอทีซีแอล (OTcl) เป็นตัวเชื่อมส่วนประกอบต่างๆ ในการจำลองการทำงาน การใช้ภาษาซีพลัสพลัส และภาษาโอทีซีแอล ซึ่งเป็นภาษาเชิงวัตถุทั้งคู่มีข้อดีคือ การนำกลับมาใช้ใหม่ และการปรับปรุงแก้ไขในภายหลัง แต่ข้อเสียคือ อาจจะทำงานได้ช้าและเปลืองหน่วยความจำ และต้องมีการวางแผนสำหรับแต่ละส่วนประกอบอย่างระมัดระวัง

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 WebCASE: A simulation environment for web caching study [2]

ในงานวิจัยนี้ได้เสนอแง่มุมของสภาพแวดล้อมในการจำลองการทำงานของเว็บแคชด้วยเว็บเคส ซึ่งเป็นเครื่องมือใช้สำหรับศึกษาการทำงานของเว็บแคช โดยเขียนขึ้นมาจากภาษาเพิร์ลเพื่อทดสอบประสิทธิภาพของขั้นตอนวิธีการที่มีอยู่แล้ว และขั้นตอนวิธีการแบบใหม่ และมีส่วนกราฟิกที่จะแสดงผลขณะเวลาทำงานด้วย แต่สภาพแวดล้อมของเว็บเคสที่ใช้ทดสอบจะเน้นขั้นตอนวิธีการในการจัดการลำดับของเอกสารที่จะเก็บภายในแคช และการแทนที่เอกสารภายในแคชของแคชเพียงตัวเดียวเท่านั้น

2.2.2 Advances in Network Simulation [10]

ในงานวิจัยนี้กล่าวว่าการเติบโตของอินเทอร์เน็ตที่เป็นไปอย่างรวดเร็วทำให้เกิดโพรโทคอลและขั้นตอนวิธีการใหม่ๆ ขึ้นมาเป็นจำนวนมาก การพัฒนาและประเมินผลเครื่องมือที่จะนำมาทดสอบจะเป็นเรื่องที่ยาก และการปรับแก้และร่วมกันใช้เครื่องมือทดสอบก็เป็นเรื่องที่ยาก และไม่ยืดหยุ่น โดยปกติการออกแบบโพรโทคอลจากการจำลองการทำงานนั่นเริ่มโดยกลุ่มทำงานกลุ่มเดียวทำการจำลองการทำงานของโพรโทคอลเดียวโดดๆ ใช้โทโพโลจีแบบธรรมดา ซึ่งการเริ่มต้นที่เสียค่าใช้จ่ายสูงนี้ ทำให้กลุ่มทำงานไม่ยอมที่จะรับเอาข้อมูลอื่นๆ มาจำลองระบบเครือข่ายที่ทันสมัยกว่าเดิม ดังนั้นแต่ละกลุ่มทำงานจึงไม่มีมาตรฐาน และการสร้างเครื่องมือใหม่ก็จะทำได้ยาก

เครื่องมือสำหรับจำลองการทำงานหลายๆ โพรโทคอลในระบบเครือข่ายช่วยให้เกิดโอกาสในการทำการทดลองที่มีประสิทธิภาพอย่างมาก ซึ่งในโปรเจควินท์ (VINT : The Virtual

InterNetwork Testbed) ได้ปรับปรุงเครื่องมือการจำลองการทำงานของระบบเครือข่ายคือ เอ็นเอส (NS) เพื่อที่นักวิจัยจะสามารถออกแบบ และใช้โพรโทคอลใหม่ๆ ได้อย่างกว้างขวางยิ่งขึ้น

2.2.3 Virtual InterNetwork Testbed: Status and Research Agenda [3]

ในงานวิจัยนี้กล่าวถึงความท้าทายในการจำลองการทำงานของระบบเครือข่ายที่เกิดขึ้นในปัจจุบัน เนื่องจากความหลากหลาย และความใหญ่โตของระบบเครือข่าย การที่จะปรับแก้การนำเข้าข้อมูลสำหรับการจำลองการทำงาน การหาแบบจำลองที่เหมาะสมในการปฏิสัมพันธ์กันระหว่างโพรโทคอลจำนวนมาก กระบวนการทำงานและผลลัพธ์ของการจำลองที่สามารถมองเห็นด้วยตานั้นทำให้ความยากยิ่งเพิ่มขึ้นไปอีก งานวิจัยนี้ยังได้อธิบายถึงการกำหนดกรอบงานในการจำลองการทำงาน การออกแบบโพรโทคอลและการปฏิสัมพันธ์กันระหว่างโพรโทคอล การสร้างสภาพแวดล้อมในการจำลองแบบต่างๆ การขยายการจำลองการทำงานและเทคนิคในการจำลองการทำงาน และผลลัพธ์ของการจำลองที่สามารถมองเห็นได้ โดยทั้งหมดจะมีความสัมพันธ์กับเอ็นเอส และนาม (NAM – Network Animator) ซึ่งเป็นซอฟต์แวร์ที่ใช้ในงานวิทยานิพนธ์นี้

2.2.4 A Survey of Web Caching Schemes for the Internet [11]

ในงานวิจัยนี้จะสำรวจรูปแบบ และวิธีการต่างๆ ของเว็บแคชที่ใช้ในอินเทอร์เน็ต เพื่อช่วยลดความคับคั่งของอินเทอร์เน็ต การทำงานที่หนักเกินไปของเซิร์ฟเวอร์ และเพิ่มความเร็วให้กับผู้ใช้ในการเข้าถึงข้อมูลด้วย ซึ่งรูปแบบและวิธีการต่างๆ ประกอบด้วย ระบบของเว็บแคช โดยแบ่งเป็น 5 แบบ คือ Harvest cache, Cachemesh system, Summary cache, Adaptive Web caching, Access driven cache สถาปัตยกรรมของการแคช แบ่งเป็น 4 แบบ คือ Hierarchical caching architecture, Distributed caching architecture, Hybrid caching architecture, Performance of caching architectures การดึงข้อมูลล่วงหน้าแบบต่างๆ เช่น ระหว่างบราวเซอร์กับเว็บเซิร์ฟเวอร์ ระหว่าง พร็อกซีกับเว็บเซิร์ฟเวอร์ และระหว่างบราวเซอร์กับพร็อกซี ขั้นตอนวิธีการในการแทนที่แบบต่างๆ เช่น LRU, Size, LRU-MIN เป็นต้น ความต้องกันของแคชทั้งแบบ Strong Cache Consistency และWeak Cache Consistency ลักษณะข้อมูลที่ทำการแคช การทำนายรูปแบบการเข้าถึงข้อมูลของผู้ใช้ การทำให้โหลดมีความสมดุล ตำแหน่งของพร็อกซี การแคชข้อมูลที่มีการเปลี่ยนแปลงอยู่ตลอดเวลา ลักษณะความคับคั่งที่เกิดขึ้นกับเว็บ ซึ่งมีรูปแบบและวิธีการบางอย่างที่ใช้ในงานวิทยานิพนธ์นี้

2.2.5 World Wide Web Cache Consistency [13]

ในงานวิจัยนี้ได้เสนอการทดสอบการทำงานของความต้องกันของเว็บแคช 3 แบบคือ Invalidate Protocol, Alex TTL (Adaptive TTL) และ TTL (Fixed TTL) โดยเปรียบเทียบแบนด์

วิดท์ เซิร์ฟเวอร์โหลด ค่ามิสเรท (Miss rate) และค่าสเทลฮิตเรท (Stale hit rate) ที่เกิดขึ้นเมื่อค่าขีดแบ่ง (Threshold) มีขนาดแตกต่างกัน ซึ่งในงานวิทยานิพนธ์นี้ได้นำผลของการเปรียบเทียบค่ามิสเรท และค่าสเทลฮิตเรท มาเปรียบเทียบกับผลการทดสอบของงานวิทยานิพนธ์นี้โดยดูจากลักษณะของกราฟที่ได้จากการทดสอบ

2.2.6 A New Efficient Caching Policy for the World Wide Web [15]

ในงานวิจัยนี้ได้เสนอขั้นตอนวิธีการแทนที่แบบใหม่ คือ MIX ซึ่งพิจารณาจากความล่าช้าของระบบเครือข่าย ขนาดของเอกสาร ความถี่ในการเรียกขอ และระยะเวลาที่ผ่านไปตั้งแต่มีการอ้างอิงถึงเอกสารภายในแคช ทำการทดสอบโดยจำลองการทำงานของแคชจากข้อมูลการใช้เว็บของ DEC [12], BU, NLANR และ INRIA เปรียบเทียบค่าฮิตเรท ไบต์ฮิตเรท และอัตราความล่าช้า ซึ่งในงานวิทยานิพนธ์นี้ได้นำผลของการเปรียบเทียบค่าฮิตเรท และไบต์ฮิตเรทระหว่างขั้นตอนวิธีการแทนที่แบบต่างๆ โดยการใช้แคชที่มีขนาดแตกต่างกัน มาเปรียบเทียบกับผลการทดสอบของงานวิทยานิพนธ์นี้โดยดูจากลักษณะของกราฟที่ได้จากการทดสอบ

2.2.7 Replacement Policies for a Proxy Cache [16]

ในงานวิจัยนี้ได้วิเคราะห์ถึงลักษณะการใช้เว็บของเว็บพร็อกซี โดยดูพารามิเตอร์ทางสถิติต่างๆ เพื่อที่จะนำมาออกแบบเป็นขั้นตอนวิธีการแทนที่เอกสารภายในแคช และได้นำเสนอขั้นตอนวิธีการแทนที่แบบใหม่ คือ LRV (Lowest Relative Value) โดยเปรียบเทียบกับขั้นตอนวิธีการแทนที่แบบ LRU และขั้นตอนวิธีการแทนที่แบบอื่นๆ เช่น FIFO และ SIZE ซึ่งในงานวิทยานิพนธ์นี้ได้นำผลของการเปรียบเทียบค่าฮิตเรท และไบต์ฮิตเรทระหว่างขั้นตอนวิธีการแทนที่แบบต่างๆ โดยการใช้แคชที่มีขนาดแตกต่างกัน มาเปรียบเทียบกับผลการทดสอบของงานวิทยานิพนธ์นี้โดยดูจากลักษณะของกราฟที่ได้จากการทดสอบ