การประชุมวิชาการระดับชาติ

มหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ **ครั้งที่ 3**

และการประชุมวิชาการระดับชาติ เครือข่ายวิจัยประชาชื่น **ครั้งที่ 4**

"บูรณาการงานวิจัย ก้าวไกลสู่ Thailand 4.0"

บทคัดย่อ

27 เมษายน 2561

ณ มหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ

ศูนย์พระนครศรีอยุธยา หันตรา

www.ruscon.rmutsb.ac.th

บทคัดย่อ

การประชุมวิชาการระดับชาติมหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ ครั้งที่ 3 และการประชุมวิชาการระดับชาติ เครือข่ายวิจัยประชาชื่น ครั้งที่ 4

“บูรณาการงานวิจัย ก้าวไกลสู่ Thailand 4.0”

วันที่ 27 เมษายน 2561
ณ มหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ ศูนย์พระนครศรีอยุธยา หันตรา
อำเภอพระนครศรีอยุธยา จังหวัดพระนครศรีอยุธยา

จัดโดย
มหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิร่วมกับเครือข่ายวิจัยประชาชื่น

หน่วยงานสนับสนุน
มหาวิทยาลัยคริสเตียน
มหาวิทยาลัยราชภัฏอุตรดิตถ์
สำนักงานคณะกรรมการวิจัยแห่งชาติ
สำนักงานคณะกรรมการการอุดมศึกษา
สำนักงานกองทุนสนับสนุนการวิจัย
กระทรวงวิทยาศาสตร์และเทคโนโลยี
เครือข่ายวิจัยประชาชื่น
เครือข่ายวิจัยเครือข่ายอุดมศึกษาภาคกลางตอนบน
เครือข่ายสหวิทยาการ ภากลาง สำนักงานราชบัณฑิตสภา

บทคัดย่อ

การประชุมวิชาการระดับชาติมหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ ครั้งที่ 3
และการประชุมวิชาการระดับชาติ เครือข่ายวิจัยประชาชื่น ครั้งที่ 4

บรรณาธิการที่ปรึกษา ผศ.ไพศาล บุรินทร์วัฒนา
รศ.ดร.กิตติ บุญเลิศนิรันดร์

บรรณาธิการผู้ทรงคุณวุฒิ
ศ.ดร.ประยุทธ อัครเอกฒาลิน
ศ.ดร.ปิติ สุคนธสุขกุล
ศ.ดร.เกตุ กรุดพันธ์
ศ.ดร.งามผ่อง คงคาทิพย์
ศ.ดร.มนต์ชัย เทียนทอง
ศ.ดร.วาสนา วงษ์ใหญ่
ศ.ดร.อรอนงค์ นัยวิกุล
ศ.ดร.อุทัยรัตน์ ณ นคร
รศ.ดร.วรากร ไม้เรียง
รศ.ดร.วันชัย ยอดสุดใจ
รศ.ดร.วิบูลย์ ชื่นแขก
รศ.ดร.บุญใจ ศรีสถิตย์นรากูร
รศ.ดร.สายวรุฬ ชัยวานิชศิริ
รศ.ดร.กิ่งพร ทองใบ
รศ.ดร.ปาจรีย์ ผลประเสริฐ
รศ.ดร.สนม ครุฑเมือง
รศ.ดร.กฤษมันต์ วัฒนาณรงค์
รศ.ดร.สมชัย ศรีนอก
รศ.ดร.สุพัฒน์ สุกมลสันต์
รศ.ดร.ศิริณา จิตต์จรัส
รศ.ดร.กานดา ว่องไวลิขิต
รศ.ดร.ทิพย์พาพร มหาสินไพศาล
รศ.ดร.โกสุม สายใจ
รศ.ดร.กัญญามน อินหว่าง
รศ.สยาม อรุณศรีมรกต
ผศ.ดร.สุจิตกัลยา มฤครัฐอินแปลง
ผศ.ดร.กิตติพงษ์ โสภณธรรมภาณ
ผศ.ดร.ธนวัต ลิมป์พาณิชย์กุล
ผศ.ดร.ศิวฤทธิ์ พงศกรรังศิลป์
ผศ.ดร.สุวิทย์ ไวยกุล
ผศ.ดร.ชนสิทธิ์ สิทธิ์สูงเนิน
ผศ.ดร.จันทร์จิรา วงษ์ขมทอง
ผศ.ดร.มารยาท โยทองยศ
ผศ.ดร.นิษฐา หรุ่นเกษม
ผศ.ดร.วารุณี อริยวิริยะนันท์
ผศ.ดร.สุบิน ยุระรัช
ผศ.ดร.ชนะศึก นิชานนท์
ผศ.ดร.นิศานาถ มั่งศิริ
ผศ.ณรงค์ศักดิ์ จักรกรณ์
ผศ.พินทุสร ปัสนะจะโน
ดร.พสิษฐ์ สุวรรณภิงคาร
ดร.เสถียรภัคณ์ มุขดี
ดร.พรเทพ รู้แผน
ดร.วารุณี ลัภนโชคดี
ดร.อภิเทพ แซ่โค้ว
ดร.ประกอบ ชาติภุกต์
ดร.เกรียงชัย รุ่งฟ้าใหม่

บรรณาธิการ รศ.นภัทร วัจนเทพินทร์

รองบรรณาธิการ ผศ.ดร.มนตรี สังข์ทอง
ผศ.ทวีศักดิ์ ศรีจันทร์อินทร์
อาจารย์วิชัย มาแสง

ฝ่ายพิสูจน์อักษรและจัดเตรียมต้นฉบับ
นางสาวสมพร วงศ์ศักดิ์
นางสาวขวัญฤทัย ภาคพิจารณ์
นางสาวพนิดา เฟื่องขจร
นางสาวอามีนะห์ ไชยธารี
นายวรงค์ บุญนิมิตร
นางสาวอ้อมใจ บุญหนุน
นางสาวน้ำทิพย์ แย้มกลีบบัว
นางสาวพรพิมล เชียรพิมาย
ว่าที่ร้อยตรีสุวินัย โสดาเจริญ
นายกฤชพล กาญจนลักษณ์

จัดพิมพ์โดย สถาบันวิจัยและพัฒนา มหาวิทยาลัยเทคโนโลยีราชมงคลสุวรรณภูมิ
60 หมู่ 3 ตำบลหันตรา อำเภอพระนครศรีอยุธยา จังหวัดพระนครศรีอยุธยา 13000
โทรศัพท์/โทรสาร 035-709097
www.rdi.rmutsb.ac.th

พิมพ์ครั้งแรก เมษายน 2561

พิมพ์ที่ บริษัท มั่งมีทวีแสน จำกัด
97 หมู่ 2 ตำบลคลองสระบัว อำเภอพระนครศรีอยุธยา
จังหวัดพระนครศรีอยุธยา 13000

3ST-P01 : การกระจายภาระงานเพื่อประมวลผลแบบกลุ่มเมฆ ด้วยวิธีการวิเคราะห์เนื้อหา

Efficient Load Balancing for Cloud Computing by Using Content Analysis

ปรีชา สมหวัง[1*] เทวิล สกุลบุญยงค์[1] ชัยวัฒน์ แดงจันทึก[2] และ จงกล จันทร์เรือง[2]

Preecha Somwang[1*] Thewin Sakunbunyong[1] Chaiwat Dangchuntuk[2]
and Jongkol Janruang[2]

บทคัดย่อ

ปัจจุบันคอมพิวเตอร์มีการเชื่อมต่อกันระหว่างเครือข่ายเพิ่มขึ้นอย่างรวดเร็ว เนื่องจากความต้องการนำไปใช้ในการบริหารจัดการและอำนวยความสะดวกในการทำงานมากขึ้น ส่งผลให้การให้บริการสารสนเทศแบบเครื่องให้บริการเพียงเครื่องเดียวไม่สามารถรองรับผู้ใช้บริการจำนวนมากได้เพียงพอ จึงจำเป็นต้องพัฒนาเป็นการให้บริการแบบกลุ่มคอมพิวเตอร์เซิร์ฟเวอร์ แต่ปัญหาพบว่าการให้บริการแบบกลุ่มคอมพิวเตอร์เซิร์ฟเวอร์เมื่อฮาร์ดแวร์มีความแตกต่างกันทำให้ยากต่อการบริหารจัดการ จึงพัฒนาสู่การให้บริการประมวลผลแบบกลุ่มเมฆ ซึ่งเป็นเทคโนโลยีเสมือนจริง สามารถเชื่อมต่อกับกลุ่มคอมพิวเตอร์เซิร์ฟเวอร์ที่มีความแตกต่างกันทั้งทางด้านฮาร์ดแวร์และซอฟต์แวร์ งานวิจัยนี้มีวัตถุประสงค์เพื่อกระจายภาระงานด้วยวิธีวิเคราะห์เนื้อหาการประมวลผลข้อมูลให้กับกลุ่มคอมพิวเตอร์เซิร์ฟเวอร์แบบกลุ่มเมฆ ด้วยวิธีการจัดสรรภาระงานแบบราวด์-โรบิน (round-robin) เพื่อแบ่งภาระงานให้เหมาะสม มีกระบวนการตรวจสอบสถานะภาระงานการประมวลผล จากการทดลองทำให้เห็นถึงประสิทธิภาพการกระจายภาระงานในการประมวลผลแบบกลุ่มเมฆ สามารถลดปัญหาการหยุดให้บริการเมื่อมีการเข้าใช้งานระบบสารสนเทศจากผู้ใช้จำนวนมาก มีความรวดเร็วในการประมวลผลข้อมูล สามารถเพิ่มและลดจำนวนเครื่องให้บริการได้ตามความต้องการ

คำสำคัญ: เซิร์ฟเวอร์ การประมวลผลแบบกลุ่มเมฆ กระจายภาระงาน ราวด์-โรบิน

Abstract

Nowadays, computer network has grown rapidly due to the demand for information technology management and facilitates greater functionality. The service provide based on a single machine can't accommodate a large enough. Therefore, a single server must change to combine for grouping server services. The problem in grouping server service is very hard of managing many devices which have difference hardware. Cloud computing is an extensive scalable computing infrastructure that to share existing resources. This paper aim to propose an efficient technique of load balance control by using content analysis in cloud computing. The propose technique that applied round-robin scheduling to computing and data storage. The results show that the proposed technique can improve to performance of load balancing in cloud computing and generate fewer false alarms.

Keywords: Server, Cloud computing, Load balancing, Round-robin

---

[1] คณะวิศวกรรมศาสตร์และสถาปัตยกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลอีสาน

[1] Faculty of Engineering and Architecture, Rajamangala University of Technology Isan

[2] คณะวิทยาศาสตร์และศิลปศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลอีสาน

[2] Faculty of Sciences and Liberal Arts, Rajamangala University of Technology Isan

[*] Corresponding author. E-mail: preecha.so@rmuti.ac.th

บทนำ

ปัจจุบันความนิยมการใช้งานอินเทอร์เน็ตเพิ่มขึ้นอย่างรวดเร็ว ทำให้การให้บริการแบบเครื่องเซิร์ฟเวอร์เพียงเครื่องเดียวไม่เพียงพอ ส่งผลให้เกิดการนำเครื่องคอมพิวเตอร์เซิร์ฟเวอร์หลายเครื่องมาจัดกลุ่มเพื่อทำงานให้บริการพร้อมกัน เทคนิคที่นิยมใช้คือการกระจายการร้องขอจากเครื่องไคลเอนต์ (Client) ไปยังเซิร์ฟเวอร์แต่ละตัว แต่ปัญหาคือเมื่อมีการร้องขอมาจากไคลเอนต์จำนวนมากพร้อมกัน ทำให้ช่วงระยะเวลาการร้องขอเพิ่มขึ้น จำเป็นต้องมีวิธีการจัดการการร้องขอที่มีประสิทธิภาพ เมื่อไคลเอนต์ร้องขอเนื้อหาเดียวกันต้องส่งไปประมวลผลที่เซิร์ฟเวอร์เดียวกัน โดยมีตัวกระจายภาระงานซึ่งเป็นเซิร์ฟเวอร์พิเศษทำหน้าที่ตัดสินใจเลือกเซิร์ฟเวอร์ที่ให้บริการร้องขอจากไคลเอนต์ โดยพิจารณาจากเนื้อหาการร้องขอและสภาพภาระงานของเซิร์ฟเวอร์ การจัดกลุ่มเครื่องเซิร์ฟเวอร์ให้บริการนิยมจัดทำในรูปแบบเทคโนโลยีเสมือนจริง เพราะสามารถแก้ไขปรับปรุงการให้บริการแล้วส่งผลกระทบต่อผู้ใช้งานน้อยที่สุด (Tirumala et al., 2016)

การพัฒนาซอฟต์แวร์ของเทคโนโลยีเสมือนจริงในอดีตยังไม่มีการใช้ประสิทธิภาพจากหน่วยประมวลผลกลางอย่างเต็มประสิทธิภาพ เมื่อเทคโนโลยีเสมือนจริงได้รับความนิยมแพร่หลายเพิ่มมากขึ้น ส่งผลให้มีการพัฒนาหน่วยประมวลผลกลางที่สามารถทำงานร่วมกับเทคโนโลยีเสมือนจริงอย่างมีประสิทธิภาพเพิ่มขึ้น (Chandrima et al., 2016) จึงทำให้เกิดการให้บริการประมวลผลข้อมูลแบบกลุ่มเมฆหรือเรียกว่าระบบคลาวด์ (Cloud) สามารถตอบสนองความต้องการผู้ใช้บริการหลากหลายรูปแบบ โดยผู้ใช้สามารถเลือกขนาดทรัพยากร ขนาดหน่วยประมวลผลหรือหน่วยความจำได้ตามต้องการ การใช้ทรัพยากรเสมือนจะถูกจัดสรรมาจากทรัพยากรระบบคอมพิวเตอร์จริง ซึ่งทำให้คอมพิวเตอร์หลักมองเห็นอุปกรณ์ที่ต่อพ่วงเสมือนมีอยู่จริง ทำให้ระบบคอมพิวเตอร์เสมือนจริงหลายระบบสามารถทำงานร่วมกันได้อย่างต่อเนื่อง ประโยชน์ในการพัฒนาการประมวลผลข้อมูลแบบกลุ่มเมฆเพื่อลดค่าใช้จ่ายสำหรับพื้นที่ในการจัดเก็บข้อมูล มีความยืดหยุ่นสูง สามารถปรับเปลี่ยนได้ง่าย สะดวกต่อการเข้าใช้ระบบสารสนเทศผ่านระบบเครือข่ายอินเทอร์เน็ต (Joonseok et al., 2017)

การให้บริการประมวลผลข้อมูลแบบกลุ่มเมฆสามารถแยกได้ 3 ลักษณะคือ การให้บริการ IaaS (Infrastructure as a Service) เป็นการให้บริการโครงสร้างพื้นฐานในส่วนของคอมพิวเตอร์เซิร์ฟเวอร์เสมือน ระบบเครือข่ายและระบบปฏิบัติการ ส่วนการให้บริการแพลตฟอร์ม PaaS (Platform as a Service) เป็นการให้บริการสภาพแวดล้อมที่เอื้อต่อการพัฒนาโปรแกรม เว็บเซิร์ฟเวอร์ ฐานข้อมูล หรือการใช้งานเฉพาะอย่าง เพื่อบริการแก่นักพัฒนาระบบ และการให้บริการ SaaS (Software as a Service) เป็นการให้บริการเฉพาะซอฟต์แวร์หรือแอพพลิเคชันที่มีไว้คอยให้บริการเท่านั้น การให้บริการประมวลผลข้อมูลแบบกลุ่มเมฆจัดว่าเป็นการให้บริการในรูปแบบเทคโนโลยีเสมือนจริง (Aarti et al., 2015)

เทคโนโลยีเสมือนจริงมีด้วยกัน 3 ประเภท คือ Full virtualization การจำลองฮาร์ดแวร์ทั้งหมดขึ้นมาใหม่ ใช้งานง่าย เพราะเป็นซอฟต์แวร์ที่ติดตั้งอยู่บนระบบปฏิบัติการหลัก สามารถติดตั้งหลายระบบปฏิบัติการภายในเครื่องเดียวกันได้ แต่ประสิทธิภาพของระบบคอมพิวเตอร์เสมือนจะต่ำการควบคุมการทำงานผ่านทาง Virtual Machine Monitor (VMM) ส่วน Para virtualization เป็นการจำลองฮาร์ดแวร์กึ่งเสมือน ต้องใช้คุณสมบัติพิเศษของหน่วยประมวลผลเพื่อจัดสรรหน่วยความจำเครื่องเวิร์ฟเวอร์ต้องทำการแก้ระบบปฏิบัติการก่อนถึงสามารถใช้ระบบคอมพิวเตอร์เสมือนได้ และ OS-level virtualization เป็นการแบ่งพาทิชันขึ้นมาบนระบบปฏิบัติการหลัก แล้วกำหนดการตั้งค่าการกระทำที่เกิดขึ้นภายในพาทิชันถูกกักไว้ภายในเท่านั้น โดยพาทิชันที่สร้างขึ้นและการตั้งค่า (Configuration) มีโครงสร้างเป็นไฟล์เหมือนกับระบบปฏิบัติการหลักทุกอย่าง งานวิจัยนี้เลือกใช้เทคโนโลยีเสมือนจริงแบบ Full virtualization ซึ่งทำให้ง่ายต่อการกระจายภาระงาน แม้เครื่องเซิร์ฟเวอร์มีระบบปฏิบัติการที่แตกต่างกัน (Anish et al., 2014) เงื่อนไขการกระจายภาระงานไปยังเซิร์ฟเวอร์โดยพิจารณาจากเนื้อหา เมื่อการร้องขอที่มีเนื้อหาเดียวกันต้องถูกส่งไปยังเครื่องเซิร์ฟเวอร์เดียวกัน สามารถตอบกลับไปยังไคลเอนต์ได้ทันทีจากข้อมูลที่อยู่ในบัฟเฟอร์ และเซิร์ฟเวอร์ทุกเครื่องต้องมีการแลกเปลี่ยนสภาพภาระงานร่วมกันเท่าที่จำเป็น โดยการใช้ข้อมูลร้องขอร่วมกันระหว่างพร็อกซี่เก็บข้อมูลและแลกเปลี่ยนกันในรูปแบบรวมแคลช (Summary

cache) ทำให้ข้อมูลมีขนาดเล็กช่วยลดจำนวนข้อมูลที่แลกเปลี่ยนกันระหว่างพร็อกซี่ และต้องมีสถานะสภาพภาระงานที่ระบุว่าเครื่องใดมีภาระงานมากหรือเครื่องใดมีภาระงานน้อย เพื่อให้เครื่องกระจายภาระงานสามารถกระจายงานให้เครื่องบริการอย่างเหมาะสม (Alireza et al., 2017) การกระจายภาระงานต้องคำนึงถึงสภาพภาระงานของเครื่องเซิร์ฟเวอร์ที่ให้บริการ สามารถปรับเปลี่ยนกระจายการร้องขอเพื่อไม่ให้เครื่องให้บริการทำงานหนักเกินไป โดยพิจารณาจากเนื้อหาการร้องขอซึ่งมีข้อมูลการร้องจัดเก็บไว้ในบัฟเฟอร์ของแต่ละเซิร์ฟเวอร์ นำข้อมูลสรุปบัฟเฟอร์ (Buffer summarization) มาแลกเปลี่ยนระหว่างเซิร์ฟเวอร์ เพื่อพิจารณาว่าเซิร์ฟเวอร์ใดมีเนื้อหาการร้องขอที่ตรงกันช่วยลดความซ้ำซ้อนในการกระจายภาระงานดังแสดงรายละเอียดใน Figure 1

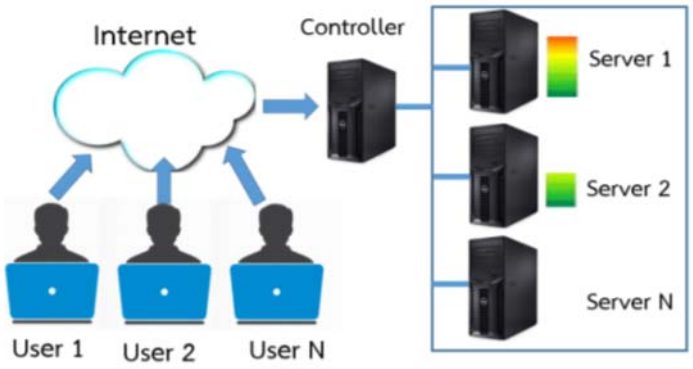


Figure 1 โครงสร้างการกระจายภาระงาน

การเพิ่มจำนวนเครื่องคอมพิวเตอร์เซิร์ฟเวอร์เพื่อจัดกลุ่มการประมวลผลด้วยเทคโนโลยีเสมือนจริง (Virtual server) สามารถทำได้ง่ายและส่งผลกระทบต่อผู้ใช้งานน้อยที่สุด โดยใช้โปรแกรม VMware ESXi สามารถแชร์ทรัพยากรให้กับเครื่องเซิร์ฟเวอร์เสมือนตามการร้องขอหรือการทำงานจริง ได้มากกว่าหนึ่งเครื่องตามทรัพยากรของเครื่องเซิร์ฟเวอร์จริงที่มีอยู่ (Abbas et al., 2015) การทำงานแสดงใน Figure 2

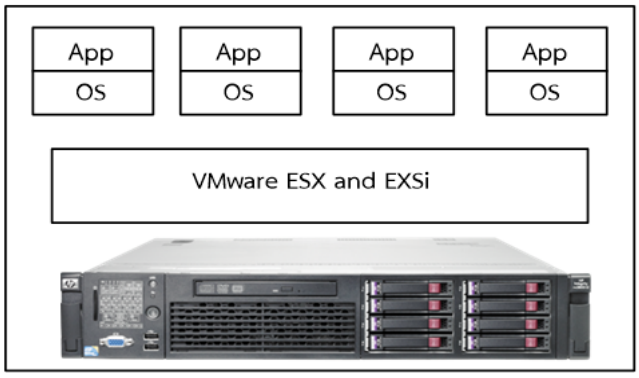


Figure 2 โครงสร้างการทำงานโปรแกรม VMware

การบริหารจัดการ VMware ESXi กรณีมีหลายเครื่อง สามารถทำได้โดยการควบคุมผ่าน VMware vCenter server สามารถย้าย Virtual server จาก ESXi server เครื่องนึงไปยังอีกเครื่องนึงได้ โดยที่ไม่ส่งผลกระทบต่อผู้ใช้งาน เมื่อจัดกลุ่มคอมพิวเตอร์เซิร์ฟเวอร์เสร็จแล้วภายในกลุ่มที่มีจำนวนเซิร์ฟเวอร์หลายตัว ต้องมีเครื่องคอมพิวเตอร์เซิร์ฟเวอร์พิเศษเครื่องหนึ่งเป็นตัวกระจายภาระงานให้กับเซิร์ฟเวอร์ทุกเครื่องภายในกลุ่ม ด้วยโปรแกรม HA proxy ด้วยการแบ่งภาระงานการประมวลผลให้กับเครื่องคอมพิวเตอร์เซิร์ฟเวอร์ที่ได้ลงทะเบียนไว้สามารถเพิ่มจำนวนเครื่องคอมพิวเตอร์เซิร์ฟเวอร์ได้

ภายหลังหากพบว่ามีจำนวนผู้ใช้งานหรือการร้องขอเพิ่มมากขึ้นความสามารถของการรองรับผู้ใช้งานเป็นจำนวนมากนั้นขึ้นอยู่กับจำนวนเครื่องคอมพิวเตอร์เซิร์ฟเวอร์ที่เปิดให้บริการ แสดงใน Figure 3

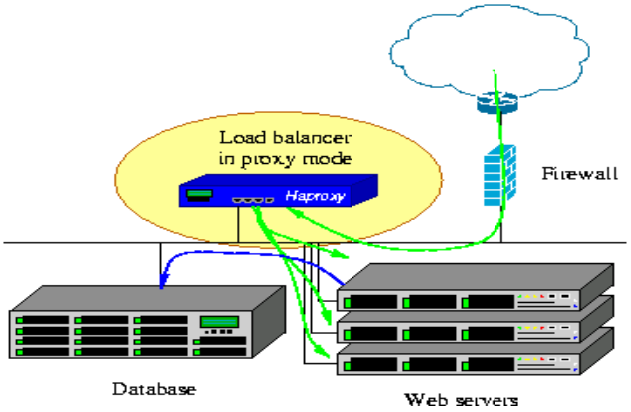


Figure 3 กระจายภาระงานด้วยโปรแกรม HA proxy

งานวิจัยนี้มีวัตถุประสงค์เพื่อกระจายภาระงานให้กลุ่มคอมพิวเตอร์ที่ทำหน้าที่เป็นเครื่องให้บริการ สำหรับประมวลผลข้อมูลแบบคลาวด์ ด้วยการจัดสรรทรัพยากรสารสนเทศแบบวนซ้ำด้วยวิธีการราวด์-โรบิน จากการวิเคราะห์เนื้อหาของการรวมแคลช (Summary cache) ทำให้ข้อมูลมีขนาดเล็กช่วยลดจำนวนข้อมูลที่แลกเปลี่ยนกันระหว่างพร็อกซี่

วิธีการศึกษา

ขั้นตอนการสร้างเครื่องคอมพิวเตอร์เซิร์ฟเวอร์เสมือนจากโปรแกรม VMware ESXi งานวิจัยนี้ใช้โปรแกรม VMware vSphere client เพื่อใช้เข้าสู้หน้าต่างการสร้างเครื่องคอมพิวเตอร์เซิร์ฟเวอร์เสมือนทำการติดตั้งโปรแกรม Vmwarev Sphere client สำหรับเข้าสู่หน้าบริหารจัดการ VMware ESXi หน้าต่างล็อกอินซึ่งต้องใส่ข้อมูล 3 ส่วนคือ ชื่อโดเมนเนมหรือหมายเลขไอพีแอดเดรสที่ต้องการเชื่อมต่อไปยังเครื่องคอมพิวเตอร์เซิร์ฟเวอร์ ESXi ชื่อผู้ใช้ และรหัสผ่านหลังจากเข้าสู่หน้าต่างบริหารจัดการเครื่องคอมพิวเตอร์เซิร์ฟเวอร์ VMware ESXi แล้วต้องสร้าง Resource pool สำหรับจัดกลุ่ม Cluster servers หลังจากนั้นก็สามารถสร้างเครื่องคอมพิวเตอร์เซิร์ฟเวอร์เสมือนตามจำนวนที่ต้องการให้บริการ เมื่อสร้างเครื่องคอมพิวเตอร์เซิร์ฟเวอร์เสมือนทั้งหมดครบแล้วขั้นตอนต่อไปเป็นการติดตั้งระบบปฏิบัติการคอมพิวเตอร์ให้กับเครื่องคอมพิวเตอร์เซิร์ฟเวอร์เสมือน (Qasim et al., 2015)

งานวิจัยนี้ใช้ระบบปฏิบัติการ Debian เป็นระบบปฏิบัติการคอมพิวเตอร์เพื่อใช้งานแบบฟรี ระบบปฏิบัติการนี้ประกอบด้วยชุดโปรแกรมพื้นฐานในด้านต่างๆ ไม่ว่าจะเป็นงานด้านเอกสาร อินเทอร์เน็ต และความบันเทิง นอกจากโปรแกรมพื้นฐานเหล่านี้ระบบปฏิบัติการ Debian ยังสามารถนำไปใช้เป็นระบบปฏิบัติการเซิร์ฟเวอร์ (Server) ให้บริการด้านเทคโนโลยีสารสนเทศได้ หลังจากติดตั้งระบบปฏิบัติการเสร็จเรียบร้อยแล้ว ขั้นตอนการล็อกอินเข้าสู่ระบบปฏิบัติการต้องใช้ชื่อผู้ใช้และรหัสผ่านที่ได้กำหนดไว้ในขั้นตอนการกำหนดผู้ใช้และรหัสผ่าน หลังจากนั้นเป็นการกำหนดค่าเครื่องคอมพิวเตอร์เซิร์ฟเวอร์ให้บริการเว็บไซต์ ด้วยระบบปฏิบัติการ Debian มีโปรแกรมที่สามารถนำมาใช้ติดตั้งได้เช่น Xammp, Apache2, Nginx, Lighttpd สำหรับการติดตั้งโปรแกรม Apache2 ได้มีการติดตั้งเข้ามาในระบบปฏิบัติการแล้วเพียงเลือกแพ็กเก็จ Web server สำหรับติดตั้ง โดยไม่ต้องมีการติดตั้งเพิ่มเติม แต่ต้องมีการอัปเกรดนอกจากโปรแกรม Apache2 แล้วโปรแกรมที่ต้องติดตั้งเพิ่มเติมสำหรับผู้พัฒนาเว็บไซต์คือโปรแกรม PHP5 จากนั้นระบบจะดาวน์โหลดโปรแกรม PHP5 พร้อมกับแพ็กเก็จสำหรับติดต่อฐานข้อมูล LDAP และ MySQL ใช้สำหรับบริการฐานข้อมูลการกระจายภาระงาน หลังจากติดตั้งเสร็จแล้วให้กำหนดการตั้งค่าไอพีแอดเดรสเพื่ออนุญาตให้เข้าใช้ฐานข้อมูลจากภายนอก

ตรวจสอบการติดตั้งโปรแกรม PHP ขั้นตอนต่อไปเป็นการทดสอบการให้บริการเว็บเซิร์ฟเวอร์ทดลองโดยเปิดเบราว์เซอร์แล้วเรียก URL เป็นหมายเลขไอพีแอดเดรส (Vincent and Bram, 2015) ในกรณีเครื่องเซิร์ฟเวอร์ภายในกลุ่มเครื่องใดเครื่องหนึ่งหยุดให้บริการ เครื่องเซิร์ฟเวอร์ตัวกระจายภาระงานสามารถส่งภาระงานเพื่อกระจายการประมวลผลไปยังเครื่องให้บริการอื่นที่ทำงานปกติ จนกว่าเครื่องที่มีปัญหากลับมาทำงานปกติเหมือนเดิม เครื่องกระจายภาระงานจึงส่งข้อมูลให้ประมวลผลใหม่อีกครั้ง (Lei Zou et al., 2017)

งานวิจัยนี้กำหนดสถานะการทำงานของเครื่องเซิร์ฟเวอร์ออกเป็น 3 สถานะคือ สภาพภาระงานน้อย สภาพภาระงานปานกลาง และสภาพภาระงานมาก โดยกำหนดให้ *L* แทนค่าสภาพภาระงานน้อย และกำหนดให้ *H* แทนค่าสภาพภาระงานมาก ดังแสดงรายละเอียดใน Figure 4

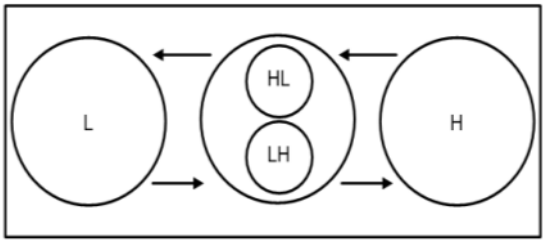
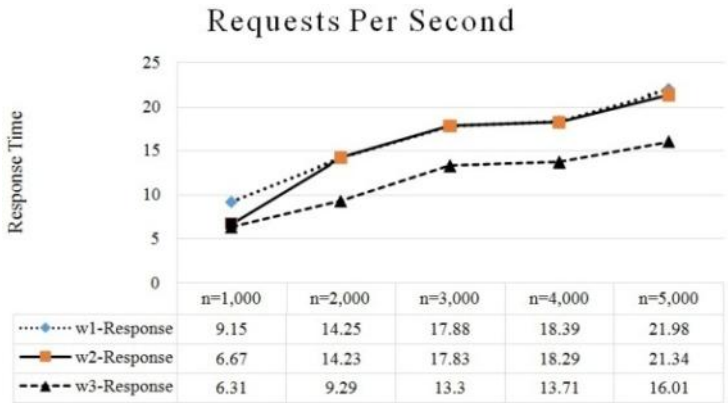


Figure 4 การแลกเปลี่ยนข้อมูลระหว่างเซิร์ฟเวอร์

เริ่มต้นด้วยสภาพภาระงานน้อยเมื่อมีการประมวลผลเพิ่มขึ้นสภานะภาระงานเปลี่ยนเป็นสภาพภาระงานปานกลาง (*LH*) ไปจนถึงภาระงานมาก ในทางกลับกันเมื่อมีภาระงานมากแล้วลดภาระงานเปลี่ยนสภาพภาระงานจากมากสู่สภาพภาระงานปานกลาง (*HL*) จนเหลือภาระงานน้อย เมื่อสภาพภาระงานของเครื่องเซิร์ฟเวอร์เปลี่ยนสถานะตัวกระจายภาระงานต้องทำการวิเคราะห์เพื่อกระจายภาระงานให้กับเครื่องเซิร์ฟเวอร์ทุกเครื่องที่อยู่ในกลุ่ม (Xiaoli and Bharadwaj, 2017)

ผลการศึกษา

การทดสอบใช้วิธีการวัดค่าการตอบสนอง (Requests per second) และค่าความผิดพลาดการร้องขอ (Failed request) (Kennedy et al., 2017) การทดสอบใช้ โปรแกรม Apache benchmark สำหรับทดสอบหาค่าการร้องขอ โดยเทียบหาจำนวนการร้องขอเว็บไซต์ต่อวินาที (Request/sec) ซึ่งได้ค่าประสิทธิภาพการตอบสนองของการบริการเว็บไซต์ผู้ใช้จำนวนมาก จำลองผู้ใช้จำนวน 100 คน ทดสอบการร้องขอจำนวน 1,000 ถึง 5,000 Request จำนวน 30 ครั้งต่อประเภทการร้องขอและหาค่าเฉลี่ยเปรียบเทียบใช้เครื่องคอมพิวเตอร์เซิร์ฟเวอร์จำลองจำนวน 1 เครื่อง (w1), 2 เครื่อง (w2) และ 3 เครื่อง (w3) ตามลำดับ ผลการทดสอบแสดงใน Figure 5



	n=1,000	n=2,000	n=3,000	n=4,000	n=5,000
w1-Response	9.15	14.25	17.88	18.39	21.98
w2-Response	6.67	14.23	17.83	18.29	21.34
w3-Response	6.31	9.29	13.3	13.71	16.01

Figure 5 อัตราการตอบสนองการร้องขอ

แสดงให้เห็นว่าเมื่อมีการร้องขอจากผู้ใช้งานจำนวนมากขึ้น จำนวนเครื่องเซิร์ฟเวอร์ให้บริการมีน้อยทำให้การตอบสนองการร้องขอใช้เวลามาก เมื่อเพิ่มจำนวนเครื่องเซิร์ฟเวอร์ให้บริการ จำนวนเวลาในการตอบสนองลดลง การเพิ่มจำนวนมากขึ้นส่งผลให้บริการรวดเร็วขึ้น นอกจากนี้การทดสอบการหาค่าความผิดพลาดการร้องขอข้อมูล (Error rate) พบว่าจำนวนเครื่องเซิร์ฟเวอร์ให้บริการเครื่องเดียวมีความผิดพลาดมากที่สุด เมื่อเพิ่มจำนวนเครื่องเซิร์ฟเวอร์เพิ่มขึ้นแนวโน้มความผิดพลาดลดลงแสดงใน Figure 6

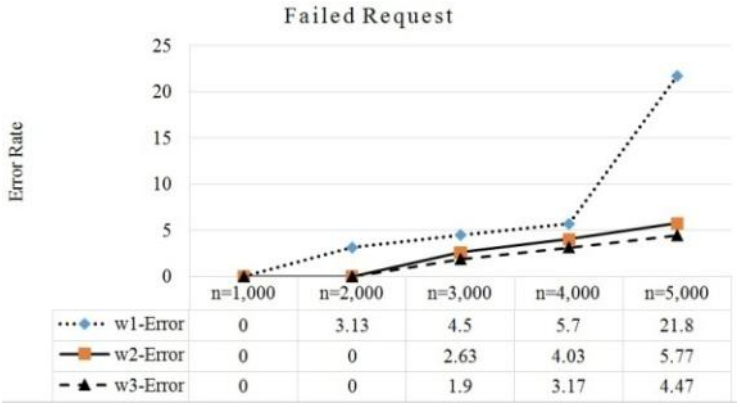


Figure 6 อัตราการตอบสนองผิดพลาด

อภิปรายผล

การกระจายภาระงานในการประมวลผลข้อมูลแบบกลุ่มเมฆ โดยการวิเคราะห์จากเนื้อหาการร้องขอของไคลเอนต์ซึ่งมีข้อมูลการร้องขอจัดเก็บไว้ในบัฟเฟอร์ของแต่ละเซิร์ฟเวอร์ นำข้อมูลสรุปบัฟเฟอร์ (Buffer summarization) มาแลกเปลี่ยนระหว่างเซิร์ฟเวอร์เพื่อกระจายภาระงานไปยังเครื่องเซิร์ฟเวอร์ให้บริการที่อยู่ภายในกลุ่มเซิร์ฟเวอร์เสมือน การทดลองทำให้ทราบถึงประสิทธิภาพการกระจายภาระงานที่เหมาะสม สามารถวิเคราะห์การร้องขอจากไคลเอนต์ได้ เมื่อเพิ่มจำนวนเครื่องเซิร์ฟเวอร์ให้บริการมากขึ้นส่งผลให้การประมวลผลข้อมูลได้รวดเร็วขึ้นและสามารถลดความผิดพลาดตามไปด้วย การเพิ่มจำนวนเครื่องเซิร์ฟเวอร์ให้บริการแต่ละครั้งไม่ส่งผลกระทบต่อการประมวลผลและการให้บริการ เมื่อเปรียบเทียบการกระจายภาระงานด้วยวิธีราวด์-โรบินกับวิธีการอื่นแสดงให้เห็นว่าวิธีราวด์-โรบินมีประสิทธิภาพมากกว่าวิธีการอื่นดังแสดงใน Figure 7

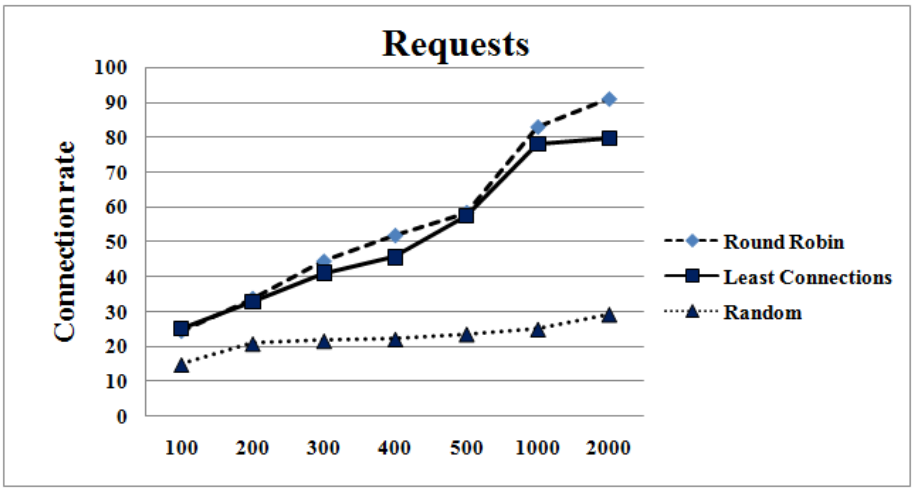


Figure 7 เปรียบเทียบประสิทธิภาพการกระจายภาระงาน

สรุป

ผลจาการทดลองทำให้ทราบถึงประสิทธิภาพการให้บริการระบบสารสนเทศด้วยวิธีการกระจายภาระงานในการประมวลผลในระบบคลาวด์ ซึ่งมีความยืดหยุ่นสามารถเพิ่มหรือลดจำนวนเครื่องให้บริการตามจำนวนผู้ใช้บริการอย่างเหมาะสมโดยไม่ส่งผลกระทบต่อการใช้งานระบบ เนื่องจากการปรับปรุงแก้ไขอยู่ในรูปแบบเทคโนโลยีเสมือนจริง (Virtualization technology) ง่ายต่อการบริหารจัดการทรัพยากรสารสนเทศของผู้ดูแลระบบ เพื่อให้การบริการมีประสิทธิภาพมากที่สุดต้องกำหนดคุณสมบัติเครื่องกระจายภาระงานให้สามารถตรวจสอบสถานะการประมวลผลและการร้องขอทรัพยากร ซึ่งมีการกระจายการเข้าถึงฐานข้อมูลสำหรับให้บริการอย่างสมดุล

เอกสารอ้างอิง

Aarti Vig, Rajendra Singh Kushwah and Shivpratap Singh Kushwah. 2015. An Efficient Distributed Approach for Load Balancing in Cloud Computing. International Conference on Computational Intelligence and Communication Networks, pp. 751 – 755.

Abbas Kiani and Nirwan Ansari. 2015. Toward Low-Cost Workload Distribution for Integrated Green Data Centers. IEEE Communications Letters, Volume: 19, Issue: 1, pp. 26 – 29.

Alireza Khoshkbarforoushha, Rajiv Ranjan, Raj Gaire, Ehsan Abbasnejad, Lizhe Wang and Albert Y. Zomaya. 2017. Distribution Based Workload Modelling of Continuous Queries in Clouds. IEEE Transactions on Emerging Topics in Computing. Volume: 5, Issue: 1, pp. 120 – 133.

Anish Babu S., Hareesh M. J., John Paul Martin, Sijo Cherian and Yedhu Sastri. 2014. System Performance evaluation of Para virtualization, Container virtualization and Full virtualization using Xen, OpenVZ and XenServer. Fourth International Conference on Advances in Computing and Communications, pp. 247 – 250.

Chandrima Dadi, Ping Yi, Zongming Fei and Hui Lu. 2016. A New Block-Based Data Distribution Mechanism in Cloud Computing. IEEE 3rd International Conference on Cyber Security and Cloud Computing, pp. 54 – 59.

Joonseok Park, Donggyu Yun, Ungsoo Kim and Keunhyuk Yeom. 2017. Approach for Cloud Recommendation and Integration to Construct User-Centric Hybrid Cloud. IEEE International Conference on Smart Cloud, pp. 24-32.

Kennedy A. Torkura, Muhammad I. H. Sukmana, Feng Cheng and Christoph Meinel. 2017. Leveraging Cloud Native Design Patterns for Security-as-a-Service Applications. IEEE International Conference on Smart Cloud, pp. 90 – 97.

Lei Zou, Zidong Wang, Huijun Gao and Xiaohui Liu. 2017. State Estimation for Discrete-Time Dynamical Networks with Time-Varying Delays and Stochastic Disturbances Under the Round-Robin Protocol. IEEE Transactions on Neural Networks and Learning Systems, Volume: 28, Issue: 5, pp. 1139 – 1151.

Qasim Ali, Haoqiang Zheng, Tim Mann and Raghunathan Srinivasan. 2015. Power Aware NUMA Scheduler in VMware's ESXi Hypervisor. IEEE International Symposium on Workload Characterization, pp. 193 – 202.

Tirumala S. S., Abdolhossein Sarrafzadeh and Paul Pang. 2016. A survey on internet usage and cyber security awareness in students. 14th Annual Conference on Privacy, Security and Trust, pp. 223 – 228.

Vincent Boisselle and Bram Adams. 2015. The impact of cross-distribution bug duplicates, empirical study on Debian and Ubuntu. IEEE 15th International Working Conference on Source Code Analysis and Manipulation, pp. 131 – 140.

Xiaoli Wang and Bharadwaj Veeravalli. 2017. Performance Characterization on Handling Large-Scale Partitionable Workloads on Heterogeneous Networked Compute Platforms. IEEE Transactions on Parallel and Distributed Systems, Volume: 28, Issue: 10, pp. 2925 – 2938.