

ตัวแบบบริการแคชสำหรับการแคชคลาวด์แบบใช้ร่วมกันที่ฝั่งไคลเอ็นท์: ทักษะเชิงเทคนิคและเชิงเศรษฐศาสตร์

CACHE-AS-A-SERVICE MODELS FOR CLIENT-SIDE SHARED CLOUD CACHING: TECHNICAL AND ECONOMICAL PERSPECTIVES

จตุรงค์ ศรีวิโรจน์

นักศึกษาระดับปริญญาโท สาขาวิชาเทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม
Email : studychr8@gmail.com

ผู้ช่วยศาสตราจารย์ ดร.เทพฤทธิ์ บัณฑิตวัฒนวงศ์

อาจารย์ประจำหลักสูตรปริญญาโท สาขาวิชาเทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม
Email : thepparit.ba@spu.ac.th

บทคัดย่อ

ในปัจจุบันคลาวด์ถูกนำมาใช้ในการจัดเก็บข้อมูลขนาดใหญ่สำหรับการใช้ร่วมกันระหว่างผู้ใช้ ทำให้องค์กรต้องมีค่าดำเนินการจากค่าโหลดข้อมูลจากคลาวด์ที่แพงและเกิดความล่าช้าในการเข้าถึงข้อมูล ปัญหาเหล่านี้สามารถแก้ไขด้วยการแคชคลาวด์เพื่อป้องกันการโหลดข้อมูลซ้ำจากคลาวด์ มีงานวิจัยล่าสุดที่เสนอกฎสำคัญของระบบแคชคลาวด์ที่ฝั่งผู้ใช้บริการคลาวด์ซึ่งสามารถประหยัดค่าโหลดข้อมูลจากคลาวด์ ลดปริมาณการใช้แบนด์วิดท์ และระยะเวลาในการโหลดข้อมูลได้ บทความวิจัยนี้ต่อยอดงานวิจัยดังกล่าวเพื่อให้สามารถเป็นบริการแคชคลาวด์ (Cache-as-a-Service) ได้โดยการเสนอตัวแบบทางเทคนิคและตัวแบบทางเศรษฐศาสตร์สำหรับการดำเนินการบริการแคชคลาวด์แบบใช้ร่วมกันที่ฝั่งไคลเอ็นท์ ที่มุ่งเน้นการลดค่าใช้จ่ายของการโหลดข้อมูลและการลดระยะเวลาการโหลดข้อมูลจากคลาวด์ การประเมินผลทั้งในด้านของสมรรถนะทางเทคนิคและสมรรถนะทางเศรษฐศาสตร์ด้วยวิธีการจำลองการทำงาน พบว่าตัวแบบบริการทางเทคนิคมีอัตราประหยัดค่าโหลดข้อมูลร้อยละ 7.40 ถึง 14.41 และพบว่าตัวแบบบริการทางเศรษฐศาสตร์แนะนำให้กำหนดราคาค่าบริการต่อเดือนโดยเฉลี่ย 16.34 ดอลลาร์สำหรับชุดข้อมูลที่ใช้ศึกษา

คำสำคัญ : การคำนวณแบบคลาวด์ การแคชคลาวด์ ตัวแบบบริการ อัตราค่าโหลดข้อมูล

ABSTRACT

Presently, cloud has been used to store large data for sharing among users imposing on organizations expensive operational costs due to cloud data loading and data access delays. These problems can be solved by using cloud caching to prevent repetitive data loading from clouds. There is a recent research study proposing the core mechanism of cloud caching at client side that can economize data loading expense, reduce bandwidth usage and data loading latency. This paper extends such research so that it becomes a cloud cache as a service (CaaS) by proposing technical and economic models for operating client-side shared CaaS, which aims for reducing

cloud data loading charges and delays. Evaluation in both technical and economic performance aspects by means of simulation has turned out that the technical service model has achieved cost-saving ratios of 7.40% to 14.41%, and the economic service model has recommended the monthly service charge of CaaS to be 16.34 dollars per month by average based on studied data sets.

KEYWORDS : Cloud computing, Cloud cache, Service model, Cost-Saving Ratio

ความเป็นมาและความสำคัญของปัญหา

การคำนวณแบบคลาวด์เป็นกระบวนการที่มีการใช้งานอย่างแพร่หลาย สถาบันมาตรฐานและเทคโนโลยีแห่งชาติสหรัฐอเมริกา (National Institute of Standards and Technology - NIST) กำหนดนิยามของการคำนวณแบบคลาวด์คือ "ตัวแบบสำหรับการทำให้การเข้าถึงแหล่งรวมทรัพยากรคอมพิวเตอร์ที่ใช้ร่วมกันและปรับแต่งโครงสร้างได้อันได้แก่ เครือข่าย, เซิร์ฟเวอร์, หน่วยเก็บ, โปรแกรมประยุกต์ และบริการ สามารถดำเนินการผ่านเครือข่ายตามคำร้องขอได้อย่างสะดวก ซึ่งทรัพยากรเหล่านี้สามารถจัดหาและคืนได้อย่างรวดเร็ว โดยอาศัยการจัดการและการโต้ตอบกับผู้ใช้บริการน้อยที่สุด" (Mell and Grance, 2011)

ในปัจจุบันคลาวด์ถูกนำมาใช้ในการจัดเก็บข้อมูลขนาดใหญ่ (Big Data) สำหรับการใช้ร่วมกันแบบกระจาย (distributed sharing) ด้วยการโหลดข้อมูลจากคลาวด์มายังผู้ใช้ปลายทางซึ่งโดยส่วนใหญ่อาศัยโพรโทคอลมาตรฐานเอชทีทีพี (HTTP) ผลที่ตามมาคือ แบนด์วิดท์ของเครือข่ายอาจเข้าสู่ภาวะอิ่มตัวและนำไปสู่ความล่าช้าในการเข้าถึงข้อมูลที่เก็บอยู่ในคลาวด์เมื่อการใช้งานข้อมูลร่วมกันมีปริมาณมาก นอกจากนี้ยังอาจเป็นสาเหตุให้องค์กรมีค่าดำเนินการจากค่าโหลดข้อมูลจากคลาวด์ที่แพงตามปริมาณการโหลดข้อมูลด้วย วิธีหนึ่งในการแก้ปัญหาเหล่านี้คือ การแคชคลาวด์ (cloud caching) ซึ่งเป็นการทำซ้ำข้อมูลที่โหลดมาจากคลาวด์และมีโอกาสถูกเรียกใช้งานซ้ำไว้ที่ฝั่งของกลุ่มผู้ใช้เพื่อป้องกันการโหลดข้อมูลเดิมจากคลาวด์ทุกครั้งที่มีการร้องขอ จากการสำรวจงานวิจัยที่เกี่ยวข้องพบว่าบริการแคชคลาวด์ส่วนใหญ่ติดตั้งอยู่ที่ฝั่งผู้ใช้บริการคลาวด์เพื่อเพิ่ม

ประสิทธิภาพการให้บริการคลาวด์ในมิติของอัตราการพบข้อมูลในแคช (hit rate) แต่มีได้มุ่งลดค่าใช้จ่ายของการโหลดข้อมูลจากคลาวด์ที่เกิดขึ้นกับผู้ใช้บริการ มีงานวิจัยล่าสุด Bandit wattanawong, Masdisomchote, and Uthayopas (2016) ที่เสนอกฎสำคัญของการพัฒนาระบบแคชคลาวด์สำหรับติดตั้งใช้งานที่ฝั่งผู้ใช้บริการคลาวด์ที่มุ่งเน้นการลดปริมาณการรับแบนด์วิดท์ และประหยัดเวลาและค่าโหลดข้อมูลจากคลาวด์

วัตถุประสงค์ของการวิจัย

บทความวิจัยนี้ต่อยอดระบบแคชคลาวด์ดังกล่าวข้างต้น โดยปรับให้สามารถทำหน้าที่เป็นบริการแคชคลาวด์ (Cache-as-a-Service) ได้ด้วยการนำเสนอองค์ความรู้ใหม่คือ ตัวแบบทางเทคนิคและทางเศรษฐศาสตร์เพื่อให้ผู้ใช้บริการสามารถให้บริการแคชคลาวด์ที่ติดตั้งอยู่ในฝั่งผู้ใช้ (client-side shared cloud cache) แก่กลุ่มผู้ใช้ขององค์การแบบมุ่งเน้นการลดค่าใช้จ่ายของการโหลดข้อมูลและลดระยะเวลาการโหลดข้อมูลจากคลาวด์ได้ ตัวแบบที่พัฒนาขึ้นรองรับการใช้เทคโนโลยีหน่วยเก็บข้อมูลแรม (RAM) หน่วยขับโซลิดสเตต (solid state drive: SSD) และหน่วยขับจานบันทึกแบบแข็ง (hard disk drive: HDD) เป็นพื้นที่แคช และสนับสนุนแนวคิดการใช้ทรัพยากรโครงสร้างพื้นฐานคลาวด์ร่วมกันระหว่างผู้ใช้ (multitenancy) เมื่อผู้ใช้บริการแคชคลาวด์นำตัวแบบนี้ไปใช้จะช่วยให้ผู้ใช้บริการสามารถกำหนดราคาบริการได้อย่างมีประสิทธิภาพและช่วยให้ผู้ใช้บริการระดับองค์การมีข้อมูลประกอบการตัดสินใจทั้งในด้านสมรรถนะและด้านงบประมาณเพื่อเลือกใช้บริการแคชคลาวด์แบบต่างๆ ที่เหมาะสมกับตนได้

งานวิจัยที่เกี่ยวข้อง

ในอดีตมีงานวิจัยหลายชิ้นที่ศึกษาเพื่อปรับปรุงประสิทธิภาพการทำงานของเว็บแคชให้ดีขึ้น Wang (1999) ทำการสำรวจรูปแบบของค้ประกอบและคุณสมบัติที่พึงประสงค์ของเว็บแคช Feldman et al. (1999) ศึกษาประสิทธิภาพการทำงานของเว็บพริ็อกซีแคชในสภาพแวดล้อมที่แบนด์วิดท์ต่างกัน Xiao et al. (2004) ศึกษาระบบแคชซึ่งพบว่ามีหลายปัจจัยที่มีผลต่อประสิทธิภาพการทำงานของเว็บแคช Bakiras and Loukopoulos (2005) พบว่าพริ็อกซีแคชเป็นขั้นตอนแรกในการลดความล่าช้าของค้ร้องขอเอชทีทีพี Liu et al. (2005) ศึกษาการทำพริ็อกซีแคชด้วยเทคนิคไดนามิกพบว่าช่วยลดภาระของเครื่องให้บริการฐานข้อมูล Ali, Shamsuddin and Ismail (2012) ระบุว่าเว็บพริ็อกซีแคชสามารถช่วยเพิ่มประสิทธิภาพของระบบเว็บและเว็บเซิร์ฟเวอร์ที่ช่วยลดเวลาตอบสนองและช่วยประหยัดแบนด์วิดท์ของเครือข่าย เห็นได้ว่างานวิจัยในหลายหัวข้อช่วยปรับปรุงเว็บพริ็อกซีแคชให้มีประสิทธิภาพมากขึ้น วัตถุประสงค์เพื่อตอบสนองการทำงานของผู้ใช้ให้ได้ข้อมูลเร็วขึ้น อย่างไรก็ตามเว็บ พริ็อกซีแคชไม่ได้ถูกออกแบบมาเพื่อรองรับลักษณะของข้อมูลจราจรในยุคการคำนวณแบบคลาวด์ ซึ่งมีขนาดใหญ่

แคชคลาวด์เป็นการใช้แคชบนคลาวด์สำหรับเก็บข้อมูลที่ส่งออกมาของบริการคลาวด์คล้ายกับพริ็อกซีแคชยกเว้นข้อมูลที่เก็บไว้ในแคชเป็นข้อมูลคลาวด์แทนข้อมูลเว็ลด์ไวด์เว็บแบบดั้งเดิม ซึ่งความแตกต่าง หลักที่สำคัญของข้อมูลคลาวด์กับข้อมูลจากเว็ลด์ไวด์เว็บแบบเดิมคือขนาดเฉลี่ยของข้อมูลในทางเทคนิค "คลาวด์แคชโดยพื้นฐานสืบทอดความสามารถของเว็บพริ็อกซีแคชแบบเดิมเริ่มตั้งแต่ข้อมูลถูกส่งมาโดยการใช้เอชทีทีพี / ทีซีพี / ไอพี โพรโทคอลสแต็คชุดเดียวกันในเว็ลด์ไวด์เว็บ" (Bandit wattanawong and Uthayopas, 2014) งานวิจัยที่เกี่ยวข้องกับแคชคลาวด์ Chiu and Agrawal (2010) ทำวิจัยมุ่งเน้นไปที่การเปรียบเทียบประสิทธิภาพกับค่าใช้จ่ายของการดูแลรักษาแคชด้วยหลายตัวแปรของโปรแกรมประยุกต์คลาวด์ เช่น ขนาดข้อมูล ขนาดแคชรวม และความคงอยู่ของข้อมูล Chiu, Shetty and Agrawal (2010) วิจัยระบบแคชคลาวด์ที่ทำงานประสานร่วมกัน สำหรับลดเวลาการประมวลผล Chockler, Laden, and Vigfusson (2010) ศึกษาและพิจารณาความท้าทายของการสร้าง

บริการแคชคลาวด์สำหรับใช้ร่วมกันที่เป็นส่วนหนึ่งของแพลตฟอร์มคลาวด์สำหรับการแคชข้อมูลที่สะดวก Kiani et al. (2011) เสนอกลยุทธ์การแคชเนื้อหาและกลไกการแคชแบบสองฝ่ายในการบริการเนื้อหาบนคลาวด์ Chockler, Laden and Vigfusson (2011) เน้นวิจัยความท้าทายต่างๆ ที่เกี่ยวข้องกับการสนับสนุนบริการแคชคลาวด์ เช่น การกำหนดมาตรวัดที่เหมาะสม ตัวแบบบริการ การจัดการประสิทธิภาพ และการใช้ทรัพยากรร่วมกันข้ามระหว่างผู้เช่า เห็นได้ว่าการแคชคลาวด์มีงานวิจัยครอบคลุมในหลายหัวข้อทั้งเรื่องตัวระบบ ขั้นตอนวิธี กลไกการแคช การสร้างและสนับสนุนบริการแคชคลาวด์ และการเปรียบเทียบประสิทธิภาพกับค่าใช้จ่าย อย่างไรก็ตามในการเสนอบริการแคชคลาวด์ให้แก่ผู้บริโภคร ผู้ให้บริการแคชคลาวด์ (cloud cache provider) จำเป็นต้องจัดเตรียมตัวแบบบริการแคชคลาวด์ (cloud cache service model) ทางเทคนิคและทางเศรษฐศาสตร์เพื่อให้ผู้บริโภครมีโอกาสพิจารณาเลือกใช้ตัวแบบที่เหมาะสมกับความต้องการ และข้อจำกัดทางด้านค่าใช้จ่ายของผู้บริโภครเองมากที่สุด และเกิดประโยชน์สูงสุด

งานวิจัยที่เกี่ยวข้องกับตัวแบบบริการแคชคลาวด์ทางด้านเทคนิค Han et al. (2012) เสนอตัวแบบบริการแคชคลาวด์ในรูปแบบ CaaS ให้เป็นบริการเพิ่มเติมจากบริการโครงสร้างพื้นฐาน (IaaS) โดยวิธีให้ผู้ให้บริการคลาวด์กำหนดค่าพื้นที่หน่วยความจำขนาดใหญ่ที่สามารถแบ่งพาร์ติชันแบบไดนามิกซึ่งพื้นที่แคชรวบรวม มาจากหน่วยความจำระยะไกลของเครื่องให้บริการที่กำหนดและสร้างต้นแบบระบบแคชยืดหยุ่นที่ใช้หน่วยความจำระยะไกลเป็นแคช (dedicated remote memory)

งานวิจัยที่เกี่ยวข้องกับตัวแบบบริการแคชคลาวด์ทางด้านเศรษฐศาสตร์ เช่น Dash, Kantere and Ailamaki (2009) เสนอตัวแบบทางเศรษฐศาสตร์ของแคชคลาวด์ที่ปรับตนเองได้ของบริการสอบถามข้อมูลทางวิทยาศาสตร์ขนาดใหญ่เป็นตัวแบบค่าใช้จ่าย (cost model) ที่คำนึงถึงการสอบถามที่ทำไปทั้งหมดและค่าใช้จ่ายที่จำเป็น เช่น แบนด์วิดท์ เครือข่าย พื้นที่ดิสก์ และเวลาซีพียู Mach and Schikuta (2011) ศึกษาวิเคราะห์ตัวแบบทางเศรษฐศาสตร์ของค่าใช้จ่ายการคำนวณแบบคลาวด์ที่รวมค่าใช้จ่ายเชิงพาณิชย์ทุกชนิดทั้งค่าใช้จ่ายที่คงที่และค่าใช้จ่ายที่ผันแปร Chockler, Laden and Vigfusson (2011) เสนอตัวแบบทางเศรษฐศาสตร์ของบริการแคชคลาวด์สำหรับค่าใช้จ่ายที่

ถ่วงดุลระหว่างประโยชน์ของลูกค้าและค่าใช้จ่ายที่ต้องจ่ายให้ผู้ให้บริการ Kantere et al. (2011) เสนอตัวแบบอุปสงค์ต่อราคา ที่ออกแบบสำหรับการแคชคลาวด์และแบบแผนการกำหนดราคาแบบพลวัตสำหรับคำสั่งดำเนินการสอบถามข้อมูลในการแคชคลาวด์ Han et al. (2012) เสนอตัวแบบทางเศรษฐศาสตร์ของการกำหนดราคาบริการแคชคลาวด์ที่สามารถถ่วงดุลระหว่างประสิทธิภาพและค่าใช้จ่ายได้อย่างมีประสิทธิภาพ อย่างไรก็ตามงานวิจัยดังกล่าวมาเป็นการวิจัยการแคชคลาวด์และตัวแบบบริการแคชคลาวด์ที่เกิดขึ้นอยู่ทางฝั่งเซิร์ฟเวอร์ (server side) ฝั่งของผู้ให้บริการคลาวด์ทั้งหมด แต่ยังมีอีกแนวความคิดหนึ่งคือวิธีการที่ผู้บริโภคริเริ่มทำการทำซ้ำข้อมูล (data replication) ที่ไหลมาจากคลาวด์ที่มีโอกาสถูกเรียกไหลซ้ำอีกมาไว้ที่พื้นที่ในฝั่งของผู้บริโภคเองที่เรียกว่าการแคชคลาวด์ฝั่งไคลเอ็นท์ (client-side cloud caching) (Banditwattanawong and Uthayopas, 2014)

งานวิจัยที่เกี่ยวข้องกับการแคชคลาวด์ฝั่งไคลเอ็นท์ Banditwattanawong and Uthayopas (2014) เสนอนโยบายการแทนที่แคชคลาวด์ฝั่งไคลเอ็นท์ (client-side cloud cache replacement policy) ด้วยวิธีใช้อับเจกต์ข้อมูลใหม่เป็นปัจจัยในการจัดลำดับความสำคัญสำหรับการตัดสินใจลบอับเจกต์ข้อมูลออกจากแคช ผลวิจัยพบว่าสามารถประหยัดแบนด์วิดท์เครือข่าย ค่าใช้จ่ายจากข้อมูลออก และเวลาการโหลดข้อมูล และนโยบายการแทนที่แคชคลาวด์ที่เสนอมีประสิทธิภาพโดยรวมดีกว่านโยบายการแทนที่ข้อมูลแบบ LRU, GDSF และ LFUDA Banditwattanawong et al. (2016) เสนอนโยบายการแทนที่แคชคลาวด์อัจฉริยะที่เรียกว่าไอ-คลาวด์ (i-Cloud) เป็นองค์ประกอบหลักของแคชคลาวด์ฝั่งไคลเอ็นท์ อย่างไรก็ตามงานวิจัยทางฝั่งไคลเอ็นท์ยังขาดการเสนอตัวแบบบริการแคชคลาวด์สำหรับฝั่งไคลเอ็นท์ Sriwiroj and Banditwattanawong (2015a) จึงเสนอตัวแบบทางเศรษฐศาสตร์บริการแคช สำหรับการแคชคลาวด์ฝั่งไคลเอ็นท์ของการกำหนดราคาค่าใช้จ่ายบริการที่พิจารณาทั้งค่าใช้จ่ายด้านเงินลงทุน และค่าใช้จ่ายด้านการดำเนินงานและอัตรากำไรที่ต้องการ เพื่อให้ผู้ให้บริการแคชคลาวด์มีคำแนะนำในการกำหนดค่าบริการและให้ผู้ให้บริการระดับองค์กรมีข้อมูลประกอบการตัดสินใจในเรื่องค่าใช้จ่ายในการเลือกใช้บริการแคชคลาวด์แบบต่างๆ ทำการวัดผลตัวแบบทางเศรษฐศาสตร์ด้วยการจำลองการทำงาน

โดยใช้ชุดข้อมูลที่มีขนาดข้อมูลอยู่ในช่วงระหว่าง 1 เมกะไบต์ถึง 5 จิกะไบต์และงานวิจัยต่อมา Sriwiroj and Banditwattanawong (2015b) ทำการวัดผลตัวแบบทางเศรษฐศาสตร์ด้วยการจำลองการทำงานโดยใช้ชุดข้อมูลลักษณะเนื้อหาดีจิทัลขนาดใหญ่ที่มีขนาดข้อมูลอยู่ในช่วงระหว่าง 4.7 จิกะไบต์ ถึง 50 จิกะไบต์

บทความวิจัยนี้ทำการสรุปตัวแบบทางเศรษฐศาสตร์ที่นำเสนอในงานวิจัยก่อนหน้าของผู้วิจัย และเสนอตัวแบบทางเทคนิคใหม่สำหรับบริการแคชคลาวด์ฝั่งไคลเอ็นท์ที่มุ่งเน้นการลดค่าใช้จ่ายของการไหลซ้ำข้อมูลและการลดระยะเวลาการโหลดข้อมูลจากคลาวด์และทำการวัดผลด้านสมรรถนะทางเทคนิคด้วยการจำลองการทำงาน

หลักการพื้นฐานของบริการแคชคลาวด์

แผนภาพที่ 1 แสดงการติดตั้งใช้งานระบบบริการแคชสำหรับการแคชคลาวด์แบบใช้ร่วมกันฝั่งไคลเอ็นท์ ระบบจะถูกติดตั้งอยู่ในพื้นที่ใกล้องค์การของผู้ใช้เพื่อทำหน้าที่เก็บสำเนาข้อมูลที่ไหลมาจากคลาวด์ระยะไกลซึ่งมีโอกาสถูกเรียกใช้ซ้ำอีกไว้ใช้ในอนาคตเมื่อมีผู้ใช้เรียกใช้ข้อมูลเดิมก็ไม่ต้องโหลดข้อมูลจากทางผู้ให้บริการคลาวด์อีกเป็นการช่วยลดการใช้แบนด์วิดท์ในเครือข่ายประหยัดค่าใช้จ่ายในการโหลดข้อมูลและช่วยให้ส่งข้อมูลไปยังผู้ใช้เร็วขึ้น อย่างไรก็ตามการนำระบบบริการแคชมาเปิดให้บริการจริงจะเป็นไปไม่ได้หากขาดตัวแบบทางเทคนิคและทางเศรษฐศาสตร์



แผนภาพที่ 1 รูปแบบการให้บริการแคช CaaS

ตัวแบบทางเศรษฐศาสตร์

ตัวแบบทางเศรษฐศาสตร์ของบริการแคชสำหรับการแคชคลาวด์ที่นำเสนอจะถูกใช้ในการกำหนดราคาค่าใช้จ่ายบริการแบบต่างๆ โดยผู้ให้บริการแคชคลาวด์ตามแผนภาพที่ 2 ตัวแบบจะอยู่บนพื้นฐานการพิจารณาให้มีความสำคัญในหลายด้านทั้งเงินลงทุน ค่าใช้จ่ายดำเนินการ และอัตรากำไรขั้นต้น ตัวแบบเกี่ยวข้องกับหลายปัจจัยค่าใช้จ่ายและหลายพารามิเตอร์ดังที่กำหนดไว้ในตารางที่ 1 ในงานวิจัยของ Sriwiroj and Banditwattanawong (2015b) ในการกำหนดตัวแบบการกำหนดราคาสำหรับบริการแคชสำหรับการแคชคลาวด์แบบใช้ร่วมกันที่ฝั่งไคลเอ็นท์จำเป็นต้องเข้าใจค่าใช้จ่ายพื้นฐานในการให้บริการ 2 ส่วนหลักคือ ส่วนแรกคือค่าใช้จ่ายในการลงทุน (CapEx) ต้นทุนที่เป็นค่าใช้จ่ายสำหรับการเริ่มต้นธุรกิจหรือระบบบริการเพื่อซื้อสินทรัพย์ถาวรมาใช้ในการดำเนินงานและค่าใช้จ่ายนี้ต้องคำนวณตัดค่าเสื่อมราคาสินทรัพย์ตลอดช่วงระยะเวลาการใช้งานหรือลดค่าลงในทุกรอบระยะเวลาขึ้นอยู่กับกฎเกณฑ์ที่กำหนด ส่วนที่สองคือค่าใช้จ่ายในการดำเนินงาน (OpEx) เป็นค่าใช้จ่ายในการดำเนินธุรกิจเช่นค่าใช้จ่ายทั่วไป และค่าใช้จ่ายในการบริหารจัดการ ซึ่งค่าใช้จ่ายนี้ประกอบด้วย 2 ส่วนคือ ต้นทุนคงที่ (OpExf) ซึ่งเป็นค่าใช้จ่ายคงที่ทุกงวดและต้นทุนผันแปร (OpExv) ซึ่งเป็นค่าใช้จ่ายตามระยะเวลาที่แตกต่างของการให้บริการ

ตัวแบบทางเศรษฐศาสตร์ของบริการแคชมาจากการคำนวณ AC คือ ค่าใช้จ่ายจริงทั้งหมดซึ่งเป็นผลรวมของ CapEx กับ OpEx บวกรวมกับ PF คือกำไรทั้งหมด ได้เป็น PR เป็นสมการพื้นฐานตัวแบบการกำหนดราคา

$$PR = AC + PF \quad (1)$$

สำหรับสมการคำนวณ CapEx จะต้องมีการคำนวณการเสื่อมราคา (depreciation) ตลอดอายุการใช้งานของแต่ละสินทรัพย์ถาวร กำหนด D_d คือค่าเสื่อมราคาต่อปีของสินทรัพย์ถาวร, d คือชุดของสินทรัพย์ถาวร (โครงสร้างพื้นฐาน, ฮาร์ดแวร์, ซอฟต์แวร์), C_d คือมูลค่าสินทรัพย์, R_d คือมูลค่าสุดท้ายที่เหลือของสินทรัพย์, p_d คือจำนวนปีที่ใช้

$$D_d = ((C_d - R_d) / p_d) / 12 \quad (2)$$

สมการ CapEx รายเดือนต่อผู้ใช้บริการหนึ่งเซิร์ฟเวอร์ กำหนด P คือจำนวนเซิร์ฟเวอร์ทั้งหมดของผู้ให้บริการ ได้สมการคือ

$$CapEx = \sum D_d / P \quad (3)$$

สมการ PFCapEx เป็นการคิดกำไรจากส่วนของ CapEx กำหนด pm คืออัตราเปอร์เซ็นต์กำไรที่ต้องการ ได้สมการ คือ

$$PFCapEx = (CapEx \times pm) / 100 \quad (4)$$

ดังนั้นสมการตัวแบบการกำหนดราคาส่วนของ CapEx คือ

$$PRCapEx = CapEx + PFCapEx \quad (5)$$

สำหรับสมการคำนวณ OpExf จะต้องอยู่ในรูปของค่าตัดจำหน่าย (amortization) ที่ซึ่งเป็นกระบวนการทางบัญชีของจำนวนเงินในระหว่างช่วงเวลา กำหนด A_{am} คือ ค่าตัดจำหน่าย, P คือ เงินต้นกู้ยืมมาลงทุน, e คือ อัตราดอกเบี้ย, q คือ จำนวนครั้งการจ่ายคืนเป็นรายเดือน, am คือ ชุดของเงินกู้ยืมสำหรับ (เงินเดือนผู้ดูแลระบบ, ค่าเช่าพื้นที่ไคลเอ็นท์เพื่อขึ้นระบบ, ค่าบำรุงรักษาฮาร์ดแวร์, ค่าบำรุงรักษาซอฟต์แวร์), ได้สมการคือ

$$A_{am} = (P_{am} \times e_{am} \times (1 + e_{am})^q) / ((1 + e_{am})^q - 1) \quad (6)$$

สมการ OpExf รายเดือนต่อผู้ใช้บริการหนึ่งเซิร์ฟเวอร์คือ

$$OpExf = \sum A_{am} / P \quad (7)$$

สมการ PFOpExf เป็นการคิดกำไรจากส่วนของ OpExf คือ

$$PFOpExf = (OpExf \times pm) / 100 \quad (8)$$

ดังนั้นสมการตัวแบบการกำหนดราคาส่วนของ OpExf คือ

$$PROpExf = OpExf + PFOpExf \quad (9)$$

สำหรับสมการคำนวณ OpExv จะต้องพิจารณาจากตัวข้อมูลที่ก่อให้เกิดค่าใช้จ่าย กำหนด o_i คือตัวข้อมูล, $Sg_{o_i}(\Delta t)$ คือ ค่าใช้จ่ายที่ข้อมูลถูกเก็บรักษาอยู่ในแคชในช่วงเวลาหนึ่ง, t_{store} คือ เวลาที่เก็บข้อมูลลงแคช, t_{evict} คือ เวลาที่ข้อมูลถูกลบออกจากแคช, Δt คือ ช่วงเวลาจาก t_{store} ถึง t_{evict} , Toi คือ ค่าไหลลดข้อมูลจากคลาวด์มาที่ระบบแคชคลาวด์, $S_{c,o_i}(\Delta t)$ คือ ค่าการใช้พื้นที่แคชในช่วงเวลาหนึ่ง สมการคำนวณค่าใช้จ่ายที่ข้อมูลถูกเก็บรักษาอยู่ในแคชคือ

$$Sg_{o_i}(\Delta t) = Toi + S_{c,o_i}(\Delta t) \quad (10)$$

สมการคำนวณค่าใช้จ่ายในส่วนของการทำแคชซึ่งกำหนด M_{p,o_i} คือ ค่าใช้จ่ายของกระบวนการที่ดูแลจัดการข้อมูล, O_{o_i} คือ ค่าใช้บริการ CaaS ในการส่งออกข้อมูลให้กับผู้ใช้

ซึ่งในการแคชจะมีตรรกะเป็น 2 กรณีคือ กรณีแรกไม่พบข้อมูลในแคชจะเรียกว่า cache-miss และกรณีพบข้อมูลในแคชจะเรียกว่า cache-hit ซึ่งสมการในกรณี cache-miss คือ

$$Cmiss_{c,p}o_i = M_p o_i + Sg_{c,p}(\Delta t) + Oo_i \quad (11)$$

สมการกำไรของกรณี cache-miss คือ Pfmisso_i ดังนี้

$$Pfmisso_i = (Cmiss_{c,p}o_i \times pm) / 100 \quad (12)$$

สมการตัวแบบการกำหนดราคาในกรณี cache-miss คือ

$$PRmiss_{c,p}o_i = Cmiss_{c,p}o_i + Pfmisso_i \quad (13)$$

และสมการในกรณี cache-hit คือ

$$Chit_{c,p}o_i = M_p o_i + Oo_i \quad (14)$$

สมการกำไรของกรณี cache-hit คือ Pfhit_i ดังนี้

$$Pfhit_i = (Chit_{c,p}o_i \times pm) / 100 \quad (15)$$

สมการตัวแบบการกำหนดราคาในกรณี cache-hit คือ

$$PRhit_{c,p}o_i = Chit_{c,p}o_i + Pfhit_i \quad (16)$$

ดังนั้นสมการตัวแบบการกำหนดราคาส่วนของ OpEx กำหนด h คือ จำนวน cache-hit ต่อเดือน, m คือ จำนวน cache-miss ต่อเดือน, j คือครั้งที่ hit, l คือครั้งที่ miss, คือ

$$PROpExv = \left(\sum_{j=1}^m PRmiss_{c,p}o_i + \sum_{j=1}^h PRhit_{c,p}o_i \right) / P \quad (17)$$

ดังนั้นสมการตัวแบบการกำหนดราคาส่วนของ OpEx คือ

$$PROpEx = PROpExf + PROpExv \quad (18)$$

และสุดท้าย สมการตัวแบบการกำหนดราคาสำหรับบริการแคชสำหรับการแคชคลาวด์แบบใช้ร่วมกันที่ฝั่งไคลเอ็นท์ในการคิดราคาค่าบริการต่อหนึ่งผู้ใช้บริการต่อเดือน คือ

$$PR = PRCapEx + PROpEx \quad (19)$$

การวัดผล

การประเมินผลตัวแบบทางเศรษฐศาสตร์ของบริการแคชคลาวด์ด้วยวิธีการจำลองการทำงานโดยใช้ชุดข้อมูลตัวแทนจากองค์การที่เน้นการใช้ข้อมูลคลาวด์ขนาดกลางซึ่งประกอบไปด้วยแฟ้มข้อมูลวิดิทัศน์ขนาด 4.7 จิกะไบต์, แฟ้มข้อมูลสื่อการเรียนรูรูปแบบเอ็มพีสาม เอ็มพีสี่ขนาดข้อมูล 300 เมกะไบต์, ภาพความละเอียดสูงรูปแบบเจพีค ทีพีพี จีพีฯ ขนาดข้อมูลระหว่าง 7 ถึง 25 เมกะไบต์และข้อมูลเอกสารคลาวด์อื่นๆ ขนาดข้อมูลประมาณ

1 เมกะไบต์ ผลลัพธ์จากตัวแบบบริการที่เลือกใช้คือ ตัวแบบ Isolated ที่ใช้หน่วยขั้วงานบันทึกแบบแข็ง (HDD) เป็นพื้นที่แคชและระบบแคชรองรับภาระงาน 288 หน่วยงาน ได้ผลลัพธ์จากตัวแบบการกำหนดราคาค่าใช้จ่ายบริการต่อเดือนที่อัตราค่าใรร้อยละ 10 เท่ากับ 35.88 ดอลลาร์ต่อเดือน (Sriwiroj and Banditwattana-wong, 2015a) และวัดผลตัวแบบด้วยการจำลองการทำงานโดยใช้ชุดข้อมูลตัวแทนจากองค์การที่มีลักษณะเน้นการใช้ข้อมูลคลาวด์ขนาดใหญ่ซึ่งประกอบด้วยเนื้อหาวิดีโอขนาดใหญ่คือ แฟ้มข้อมูลวิดิทัศน์ขนาดใหญ่ความละเอียดสูงขนาดข้อมูล 50 จิกะไบต์, แฟ้มข้อมูลการทำให้เคลื่อนไหวขนาดข้อมูล 20 จิกะไบต์, เนื้อหาการเรียนการสอนผ่านทางสื่ออิเล็กทรอนิกส์ขนาดข้อมูล 15 จิกะไบต์, แฟ้มข้อมูลวิดิทัศน์ความละเอียดมาตรฐานขนาดข้อมูล 4.7 จิกะไบต์ ผลลัพธ์จากตัวแบบบริการที่เลือกใช้คือตัวแบบ Isolated ที่ใช้หน่วยขั้วงานบันทึกแบบแข็ง เป็นพื้นที่แคชและมีหน่วยงานที่ใช้บริการระบบแคชจำนวน 50 หน่วยงาน ได้ผลลัพธ์จากตัวแบบการกำหนดราคาค่าใช้จ่ายบริการต่อเดือนที่อัตราค่าใรร้อยละ 10 เท่ากับ 42.34 ดอลลาร์ต่อเดือน (Sriwiroj and Banditwattana-wong, 2015b)

ตัวแบบทางเทคนิค

การให้บริการแคชคลาวด์ยังจำเป็นต้องจัดเตรียมตัวแบบทางเทคนิคของบริการแคชที่มีความยืดหยุ่นเพื่อให้ผู้ใช้มีทางเลือกที่ตรงกับความต้องการด้านสมรรถนะของผู้ใช้ ตัวแบบทางเทคนิคของบริการแคชสำหรับการแคชคลาวด์แบบใช้ร่วมกันที่ฝั่งไคลเอ็นท์ที่เสนอจะเกิดจากองค์ประกอบ 2 มิติคือ มิติแรกหมายถึงศักยภาพของเทคโนโลยีการจัดเก็บที่ใช้เป็นพื้นที่แคชที่มีแตกต่างกันของเวลาในการเข้าถึงข้อมูลคือ แรม หน่วยขั้วโซลิดสเตท และหน่วยขั้วงานบันทึกแบบแข็งและมิติที่สองหมายถึงลักษณะการใช้งานซึ่งเป็นทางเลือกระหว่างการใช้อินเทอร์เน็ตที่แคชร่วมกันข้ามกลุ่มผู้ใช้ (Shared Cache space) ซึ่งการใช้พื้นที่แคชร่วมกันจะทำให้เกิดการประหยัดการใช้ทรัพยากรและใช้พลังงานไฟฟ้าได้อย่างคุ้มค่าเหมาะกับหน่วยงานที่อยู่ภายใต้องค์การเดียวกันหรือมีความร่วมมือกันทางธุรกิจหรือทางเลือกที่ผู้ใช้บริการกำหนดใช้อินเทอร์เน็ตเฉพาะตน (Isolated Cache space) เนื่องจากต้องการความเป็นส่วนตัวของข้อมูล โดยการบูรณาการ 2 มิติข้างต้นทำให้สามารถเสนอตัวแบบทางเทคนิคได้ 6 ตัวแบบ ดังแสดงตามแผนภาพที่ 2

RAM	1. RAM Isolated Cache space	2. RAM Shared Cache space
SSD	3. SSD Isolated Cache space	4. SSD Shared Cache space
HDD	5. HDD Isolated Cache space	6. HDD Shared Cache space

Isolated Cache space Shared Cache space

แผนภาพที่ 2 ตัวแบบทางเทคนิคของบริการแคชคลาวด์

ตัวแบบบริการ RAM Isolated Cache space เหมาะกับผู้ใช้ที่ต้องการความเร็วในการเข้าถึงและการถ่ายโอนข้อมูลสูงที่สุดและให้ความสำคัญกับความปลอดภัยของข้อมูลและความเป็นส่วนตัวของข้อมูล ตัวแบบบริการ RAM Shared Cache space เหมาะกับผู้ใช้ที่ต้องการความเร็วในการเข้าถึงและการถ่ายโอนข้อมูลสูงที่สุดและองค์กรมีลักษณะที่หน่วยงานในสังกัดที่สามารถใช้พื้นที่แคชข้อมูลร่วมกันได้ เช่น ธุรกิจขนาดใหญ่ที่มีหลายสาขา

ตัวแบบบริการ SSD Isolated Cache space เหมาะกับผู้ใช้องค์กรที่ต้องการความเร็วในการเข้าถึงและการถ่ายโอนข้อมูลที่สูงและมีความเป็นส่วนตัวของข้อมูลมีงานที่ต้องโหลดข้อมูลใหม่ขนาดใหญ่ปริมาณมากอยู่เรื่อยๆ แต่มีงบประมาณปานกลาง เช่น ธุรกิจขนาดกลาง

ตัวแบบบริการ SSD Shared Cache space เหมาะกับผู้ใช้องค์กรที่ต้องการความเร็วในการเข้าถึงและการถ่ายโอนข้อมูลที่สูงและองค์กรมีลักษณะที่หน่วยงานภายใต้สามารถใช้พื้นที่แคชข้อมูลร่วมกันได้แต่ยังมีข้อจำกัดเรื่องงบประมาณระดับหนึ่ง เช่น สถาบันการศึกษา

ตัวแบบบริการ HDD Isolated Cache space เหมาะกับผู้ใช้องค์กรที่ต้องการความเร็วในการเข้าถึง และการถ่ายโอนข้อมูลสูงระดับมาตรฐานปกติและต้องการความเป็นส่วนตัวของข้อมูลและความปลอดภัยของข้อมูลมีงานที่ต้องโหลดข้อมูลใหม่ปริมาณมากขนาดใหญ่แต่มีงบประมาณน้อย เช่น ธุรกิจขนาดเล็ก

ตัวแบบบริการ HDD Shared Cache space เหมาะกับผู้ใช้องค์กรที่ต้องการความเร็วในการเข้าถึงและการถ่ายโอนข้อมูลที่สูงระดับมาตรฐานปกติและองค์กรมีลักษณะที่หน่วยงานภายใต้หลายหน่วยงานสามารถใช้พื้นที่แคชข้อมูลร่วมกันได้และมีงานที่ต้องโหลดข้อมูลปริมาณมากขนาดใหญ่มีงบประมาณน้อย เช่น องค์กรภาครัฐ สำหรับค่าบริการหรือค่าใช้จ่ายของตัวแบบบริการอธิบายในหัวข้อตัวแบบทางเศรษฐศาสตร์

การวัดผล

การประเมินผลตัวแบบทางเทคนิคของบริการแคชคลาวด์ใช้วิธีการจำลองการทำงานเทียบกับการใช้บริการบนคลาวด์โดยตรง ผลลัพธ์เชิงประสิทธิภาพใช้ตัววัดสมรรถนะทางเทคนิค ได้แก่ อัตราประหยัดค่าไหลดข้อมูล (Cost-saving ratio) และอัตราประหยัดเวลาไหลดข้อมูล (Delay-saving ratio), อัตราการพบข้อมูลในแคช (Hit rate) และอัตราขนาดรวมของข้อมูลที่พบในแคช (Byte-hit rate) ชุดข้อมูลที่ใช้คือชุดคำร้องขอการไหลดข้อมูลเว็บในอดีตบนอินเทอร์เน็ตผ่านเว็บแคชของโครงการ IRCache (NSF and NLNR, 2015) ในประเทศสหรัฐอเมริกาเป็นข้อมูลนำเข้ามีจำนวน 2 ชุด ข้อมูลแต่ละชุดถูกรวบรวมมาจากเว็บแคชในเมือง Boulder (BO) และ New York (NY) โดยทั้ง 2 ชุดข้อมูลถูกรวบรวมไว้ 31 วัน แต่ละรายการคำร้องขอข้อมูลประกอบไปด้วยข้อมูลปัจจัยที่จำเป็นต่อการจำลองการทำงานดังนี้ เวลาที่ร้องขอข้อมูล, แหล่งที่อยู่ของข้อมูลในอินเทอร์เน็ต, ระยะเวลาของการไหลดข้อมูล ซึ่งเป็นผลรวมของระยะเวลาแท้จริงในการสร้างการเชื่อมต่อและการส่งข้อมูลผ่านเครือข่ายจากเครื่องให้บริการมายังแคช, ขนาดของข้อมูลที่ร้องขอ, และเวลาที่ข้อมูลหมดอายุใช้งาน การสร้างโปรแกรมจำลองการทำงานด้วยภาษาจาวาและจาวารันไทม์ (Java(TM) SE Runtime Environment) รุ่น 1.8.0_20-b26 ที่สามารถนำเข้าชุดข้อมูลและประมวลผลการแคชในสถานะแวดล้อมแบบคลาวด์ที่มีการกำหนดค่าตัวแปรที่เกี่ยวข้องทั้งหมด สำหรับข้อกำหนดเครื่องคอมพิวเตอร์ที่ใช้จำลองการทำงานเป็นหน่วยประมวลลอินเทลแกน i7-3667U ซีพียู 2.00 จิกะเฮิร์ตซ์ หน่วยความจำหลักขนาด 6 จิกะไบต์ ระบบปฏิบัติการวินโดวส์ 7 รุ่น 64 บิต

ตัวแปรต้น

การทำการจำลองการทำงานและวัดสมรรถนะมีหลายตัวแปรต้นที่เกี่ยวข้องดังนี้ ขนาดของข้อมูลที่ร้องขอ, เวลาที่ร้องขอข้อมูล, ระยะเวลาของการไหลดข้อมูล, ความถี่ในการเข้าถึงข้อมูล, เวลาแฝงการเข้าถึงข้อมูล, การเข้าถึงข้อมูลครั้งล่าสุด, อายุใช้งานคงเหลือของข้อมูล (TTL), ขนาดพื้นที่เก็บข้อมูลของแคช, เวลาแฝงการเข้าถึงแคช, จำนวนองค์กรทั้งหมดที่ใช้บริการ, อัตราค่าไหลดข้อมูลจากผู้ให้บริการคลาวด์ (data-out charge rate), อัตราค่าไหลดข้อมูลจากผู้ให้บริการแคชคลาวด์ (Caas

data-out charge rate), ค่าใช้จ่ายในการใช้พลังงานไฟฟ้า และค่าเช่าพื้นที่โคโลเซิร์ฟเวอร์, ค่าใช้จ่ายของกระบวนการที่ดูแลจัดการข้อมูล (Mp), ค่าการใช้พื้นที่แคช (Sc), อัตรากำไรร้อยละ, ก่อนดำเนินการจำลองการทำงานเพื่อศึกษาสมรรถนะการทำงาน และวัดผลประสิทธิภาพของตัวแบบจะต้องดำเนินการขั้นตอน การกำหนดค่าตัวแปรต้นก่อนอธิบายในหัวข้อการกำหนดค่า การจำลองการทำงาน

ตัวแปรตาม

อัตราประหยัดค่าโหลดข้อมูล = (ค่าโหลดข้อมูลรวมของกรณีที่ไม่มีแคชคลาวด์ - ค่าโหลดข้อมูลรวมของกรณีที่มีแคชคลาวด์) / (ค่าโหลดข้อมูลรวมของกรณีที่ไม่มีแคชคลาวด์) กำหนด c_i คือ ค่าโหลดข้อมูลจากผู้ให้บริการคลาวด์, i คือ ข้อมูลหนึ่งหน่วย, c_k คือ ค่าโหลดข้อมูลจากบริการแคชคลาวด์, k คือ หน่วยเก็บข้อมูลแคชได้แก่ {ram, ssd, hdd}, r_i คือ จำนวนครั้งของการร้องขอข้อมูล i , s_i คือ ขนาดของข้อมูล i , h_i คือ จำนวนครั้งที่พบข้อมูล i ในแคช, m_i คือ จำนวนครั้งที่ไม่พบข้อมูลในแคช, n คือจำนวนตัวข้อมูล, ค่าโหลดข้อมูลรวมของกรณีที่ไม่มีแคชคลาวด์ = ค่าโหลดข้อมูลทั้งหมดจากคลาวด์ = $\sum_{i=1}^n c_i s_i r_i$, ค่าโหลดข้อมูลรวมของกรณีที่มีแคชคลาวด์ = ค่าโหลดข้อมูลที่ cache hit + ค่าโหลดข้อมูลที่ cache miss = $\sum_{i=1}^n c_k s_i h_i + \sum_{i=1}^n (c_i + c_k) s_i m_i$, ได้สมการคำนวณอัตราประหยัดค่าโหลดข้อมูลคือ $[\sum_{i=1}^n c_i s_i r_i - [\sum_{i=1}^n c_k s_i h_i + \sum_{i=1}^n (c_i + c_k) s_i m_i]] / [\sum_{i=1}^n c_k s_i r_i]$ อัตราประหยัดเวลาโหลดข้อมูล = (เวลาโหลดข้อมูลรวมของกรณีที่ไม่มีแคชคลาวด์ - เวลาโหลดข้อมูลรวมของกรณีที่มีแคชคลาวด์) / (เวลาโหลดข้อมูลรวมของกรณีที่ไม่มีแคชคลาวด์) กำหนด d_i คือระยะเวลาการโหลดข้อมูล i จากผู้ให้บริการคลาวด์มายังแคช, d_k คือระยะเวลาการโหลดข้อมูล i จากระบบแคชคลาวด์ไปยังผู้ใช้บริการ, เวลาโหลดข้อมูลรวมของกรณีที่ไม่มีแคชคลาวด์ = เวลาโหลดข้อมูลทั้งหมดจากคลาวด์ = $\sum_{i=1}^n d_i r_i$, เวลาโหลดข้อมูลรวมของกรณีที่มีแคชคลาวด์ = เวลาโหลดข้อมูลที่ cache hit + เวลาโหลดข้อมูลที่ cache miss = $\sum_{i=1}^n d_k h_i + \sum_{i=1}^n (d_i + d_k) m_i$, ได้สมการคำนวณอัตราประหยัดเวลาโหลดข้อมูลคือ

$[\sum_{i=1}^n d_i r_i - [\sum_{i=1}^n d_k h_i + \sum_{i=1}^n (d_i + d_k) m_i]] / [\sum_{i=1}^n d_i r_i]$ อัตราการพบข้อมูลในแคช (Hit rate) เป็นอัตราส่วนของจำนวนครั้งทั้งหมดที่พบข้อมูลในแคชต่อจำนวนการร้องขอข้อมูลทั้งหมด สมการคำนวณคือ $\sum_{i=1}^n h_i / \sum_{i=1}^n r_i$ (Podlipnig and Böszörményi, 2003), อัตราขนาดรวมของข้อมูลที่พบในแคช (Byte hit rate) เป็นอัตราส่วนขนาดรวมของข้อมูลที่ไหลออกจากแคชต่อขนาดรวมของข้อมูลที่ร้องขอทั้งหมด สมการคำนวณ คือ $\sum_{i=1}^n s_i h_i / \sum_{i=1}^n s_i r_i$ (Podlipnig and Böszörményi, 2003)

การกำหนดค่าการจำลองการทำงาน

ค่าตัวแปรต้นที่ใช้ในการจำลองการทำงานถูกกำหนดดังมีรายละเอียดต่อไปนี้

- กำหนดค่าตัวแปรต้นนโยบายการแทนที่ข้อมูล คือ ไอ-คลาวด์ (Banditwattanawong et al., 2016) ในการจำลองการทำงานประสิทธิภาพของการแคชขึ้นอยู่กับกลไกสำคัญใช้ควบคุมการลบข้อมูลออกจากแคชเมื่อพื้นที่เก็บข้อมูลในแคชไม่เพียงพอสำหรับการบันทึกข้อมูลใหม่ที่แคชไหลมา สำหรับตัวแปรต้นความถี่ในการเข้าถึงข้อมูล, เวลาแฝงการเข้าถึงข้อมูล, การเข้าถึงข้อมูลครั้งล่าสุด, อายุใช้งานคงเหลือของข้อมูลเหล่านี้เป็นตัวแปรต้นสำหรับขั้นตอนวิธีของนโยบายการแทนที่ข้อมูลในแคช
- กำหนดค่าตัวแปรต้นประเภทบริการที่มีตัวเลือกเป็นแบบแยกเดี่ยว (Isolated) หรือแบบใช้ร่วมกัน (Shared)
- กำหนดค่าตัวแปรต้นอัตราค่าโหลดข้อมูลจากผู้ให้บริการคลาวด์เป็นแบบหลายอัตรา (Nonuniform costs) ใช้ 0.0829 ดอลลาร์ต่อจิกะไบต์ของ Amazon และ 0.1535 ดอลลาร์ต่อจิกะไบต์ของ Google (Banditwattanawong, Masdisornchote and Uthayopas, 2016)
- กำหนดค่าตัวแปรต้นประเภทแคชที่มีตัวเลือกเป็นแรม หรือโซลิดสเตทหรือหน่วยขับเคลื่อนแบบแข็ง
- กำหนดค่าตัวแปรต้นเวลาแฝงการเข้าถึงแคช โดยแรมมีเวลาแฝงการเข้าถึงประมาณ 10 ถึง 15 นาโนวินาที (Goering, 2011) ส่วนโซลิดสเตทมีเวลาแฝงการเข้าถึง 0.1 มิลลิวินาทีและหน่วยขับเคลื่อนแบบแข็งมีเวลาแฝงการเข้าถึงประมาณ 10 ถึง 12 มิลลิวินาที (Samsung, 2015)

- ลักษณะของข้อมูลนำเข้าชุดข้อมูล BO มีจำนวนคำร้องขอข้อมูลเท่ากับ 639,187 รายการและมีขนาดข้อมูลที่ร้องขอทั้งหมด 2,262,144,480 ไบต์ ส่วนชุดข้อมูล NY มีจำนวนคำร้องขอข้อมูลเท่ากับ 1,311,880 รายการและมีขนาดข้อมูลที่ร้องขอทั้งหมด 10,801,010,237 ไบต์ ชุดข้อมูล BO และชุดข้อมูล NY แต่ละชุดใช้ในการจำลองการทำงานแบบ Isolated สำหรับชุดข้อมูล BONY ใช้ในการจำลองการทำงานแบบ Shared

- กำหนดค่าตัวแปรต้นขนาดพื้นที่เก็บข้อมูลของแคชของชุดข้อมูล BO, NY และ BONY ของ 31 วันโดยแต่ละชุดข้อมูลเริ่มจากร้อยละ 100 ซึ่งหมายถึงของขนาดแคชที่ไม่ทำให้เกิดการแคชพลาด (cache miss) เลย เมื่อได้ค่าขนาดพื้นที่แคชหน่วยไบต์ที่ร้อยละ 100 จึงคำนวณหาร้อยละ 30 ของขนาดแคชซึ่งใช้ในการจำลองการทำงานตามตารางที่ 1

ตารางที่ 1 ขนาดของแคชของแต่ละชุดข้อมูล

ชุดข้อมูลและร้อยละของขนาดแคช	พื้นที่แคช (ไบต์)
BO 30%	678,643,344
NY 30%	3,240,303,071
BONY 30%	3,906,694,572
BO 100%	2,262,144,480
NY 100%	10,801,010,237
BONY 100%	13,022,315,239

- การวัดเปรียบเทียบสมรรถนะของคอมพิวเตอร์ส่วนนี้ทำก่อนการกำหนดค่าของตัวแปรต้นค่าใช้จ่ายในการใช้พลังงานไฟฟ้าและตัวแปรต้นค่าใช้จ่ายของกระบวนการที่ดูแลจัดการข้อมูล (Mp_{oi}) และตัวแปรต้นจำนวนองค์การทั้งหมดที่เครื่องให้บริการสามารถรองรับได้ เพื่อหาอัตราส่วนสมรรถนะของเครื่องคอมพิวเตอร์ที่ให้บริการแคชคลาวด์เพื่อใช้อัตราส่วนสมรรถนะคำนวณในขั้นตอนการกำหนดค่าของตัวแปรต้นดังกล่าวข้างต้น เนื่องจากเครื่องคอมพิวเตอร์ที่ใช้จำลองการทำงานที่มีข้อกำหนดเป็นหน่วยประมวลผลอินเทลแกน i7-3667U จึงต้องคำนวณเทียบตรรกะประสิทธิภาพประเมินผลคะแนนการทำงานของซีพียู (CPUmark) กับเครื่องคอมพิวเตอร์ที่ใช้ในการให้บริการแคชคลาวด์ที่มีข้อกำหนดเป็นหน่วยประมวลผล 64 แกนเอเอ็มดี

รุ่นออฟเทอร์อนแบบจำลอง 6320 2.8 จิกะเฮิร์ตซ์ แคช 16 เมกะไบต์ ซึ่งใช้แค่ 56 แกนอีก 8 แกนใช้สำหรับการบริหารจัดการระบบ ผลการคำนวณได้สมรรถนะของซีพียูหน่วยประมวลผลเอเอ็มดีออฟเทอร์อนมากกว่าอินเทล i7-3667U เท่ากับ 18.18435754189944 เท่า

- กำหนดค่าตัวแปรต้นค่าใช้จ่ายในการใช้พลังงานไฟฟ้าของซีพียูหน่วยประมวลผลเอเอ็มดี 115 วัตต์ ใช้ค่าไฟฟ้าประเภทที่ 3 กิจการขนาดกลาง 2.8095 บาทต่อหน่วย (ERC, 2011) และใช้อัตราแลกเปลี่ยน 1 ดอลลาร์ เท่ากับ 35.3950752 บาท ณ วันที่ 11 สิงหาคม 2015 คำนวณค่าไฟต่อชั่วโมงและการใช้ประโยชน์ร้อยละ 25 ของซีพียูหน่วยประมวลผลได้เท่ากับ 6.3390116e-10 ดอลลาร์ต่อ 1 มิลลิวินาที

- กำหนดค่าตัวแปรต้นค่าใช้จ่ายของกระบวนการที่ดูแลจัดการข้อมูล (Mp_{oi}) ของซีพียูหน่วยประมวลผลเอเอ็มดีจากประเภทบริการแบบแยกเดี่ยวและแบบใช้ร่วมกัน และประเภทอัตราค่าโหลดข้อมูลจากผู้ให้บริการคลาวด์แบบหลายอัตราโดยการคิดจากค่าเฉลี่ยของค่าใช้จ่ายในการใช้พลังงานไฟฟ้าของซีพียูประมวลผลต่อ 1 o_i ตามตารางที่ 2

ตารางที่ 2 ค่าใช้จ่ายของ (Mp_{oi})

ประเภทและแบบอัตราค่าโหลด	ค่า Mp_{oi} (ดอลลาร์)
Isolated Nonuniform costs	4.349841e-11
Shared Nonuniform costs	9.6864047e-11

- กำหนดค่าตัวแปรต้นจำนวนองค์การทั้งหมดที่ใช้บริการทั้งหมดที่เครื่องให้บริการที่เป็นซีพียูหน่วยประมวลผลเอเอ็มดีออฟเทอร์อนสามารถรองรับได้ โดยอ้างอิงจากประสิทธิภาพของเครื่องที่ใช้จำลองการทำงานที่เป็นซีพียูอินเทล i7-3667U โดยคำนวณความสามารถในการประมวลผลจำนวนรายการต่อมิลลิวินาทีหารด้วยจำนวนรายการที่ผู้ใช้ร้องขอต่อมิลลิวินาทีทำการปรับเทียบสมรรถนะเป็นซีพียูเอเอ็มดีออฟเทอร์อน ได้ผลเป็นจำนวนองค์การทั้งหมดที่เครื่องให้บริการซีพียูเอเอ็มดีสามารถรองรับได้โดยเฉลี่ยเท่ากับ 118 องค์การ

- กำหนดค่าตัวแปรต้นค่าการใช้พื้นที่แคช Sc_{oi} ซึ่งคำนวณหาจากขนาดความจุของแคช ราคาซื้อ และการใช้

ประโยชน์เต็มที่ 3 ปี ได้เป็นราคาการใช้พื้นที่แคชหน่วยดอลลาร์ต่อ 1 จิกะไบต์ต่อ 1 มิลลิวินาที ตามตารางที่ 3

ตารางที่ 3 ค่าการใช้พื้นที่แคช (Sc₀)

หน่วยเก็บข้อมูล	ค่า S _{coi} (ดอลลาร์ต่อจิกะไบต์ต่อมิลลิวินาที)
RAM	1.1495429e-10
SSD	9.1032991e-12
HDD	1.1346888e-12

- กำหนดค่าตัวแปรต้นค่าเช่าพื้นที่โคโลเซิร์ฟเวอร์ (KSC, 2015) แบบ 1U ราคา 3000 บาทต่อเดือน เข้าใช้ 2 หน่วย 1 ปี ใช้อัตราแลกเปลี่ยน 1 ดอลลาร์ เท่ากับ 35.3950752 บาท ณ วันที่ 11 สิงหาคม 2015 ได้เท่ากับ 2034.18129762866 ดอลลาร์
- กำหนดค่าตัวแปรต้นอัตราค่าร้อยละ 3.8 (Foong and Delcroix, 2015)

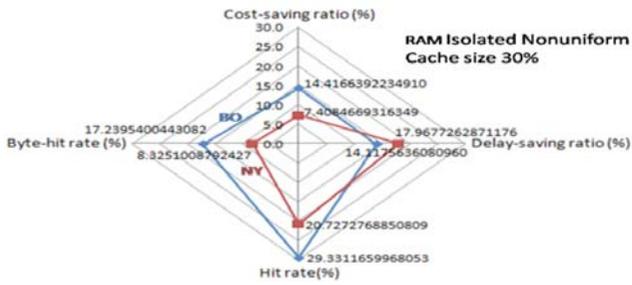
- กำหนดค่าตัวแปรต้นอัตราค่าโหลดข้อมูลจากผู้ให้บริการแคชคลาวด์ โดยดำเนินการจำลองการทำงานของระบบแคชคลาวด์ ด้วยชุดข้อมูลที่อ้างอิงจากงานวิจัยของ Sriwiroj and Banditwattanawong (2015b) และด้วยค่าตัวแปรต้นที่กำหนดดังที่กล่าวมาของเครื่องผู้ให้บริการซีพียูหน่วยประมวลผลเอ็มดี ออฟเทอร์อน โดยไม่คิดอัตราค่าโหลดข้อมูล ทำการจำลองการทำงานจนครบทุกกรณีและทำการคำนวณจนครบทุกกรณี ได้อัตราค่าโหลดตามตารางที่ 4

ตารางที่ 4 อัตราค่าโหลดข้อมูลจากผู้ให้บริการแคชคลาวด์ CaaS Data-Out Charges Rate

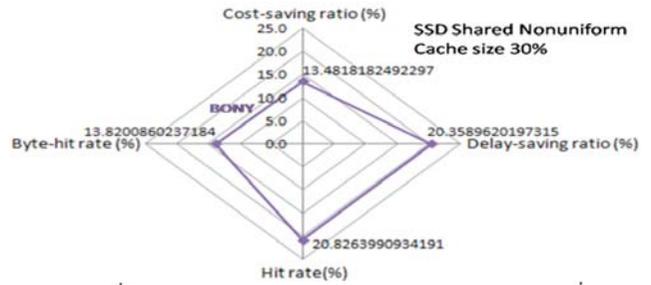
ชนิดแคช	แบบหลายอัตรา	
	แบบแยกเดี่ยว	แบบใช้ร่วมกัน
RAM	4.22656811026883e-4	4.22656811134401e-4
SSD	4.22656810990041e-4	4.22656811052360e-4
HDD	4.22656810987267e-4	4.22656811046184e-4

ผลลัพธ์และกราฟ

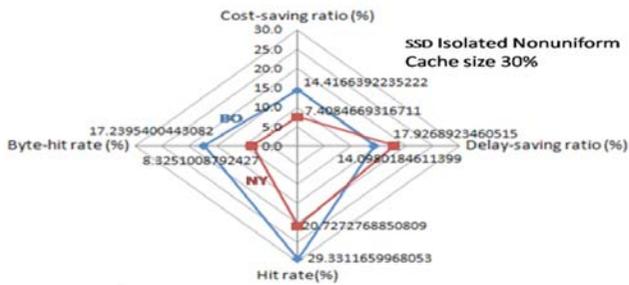
ผลลัพธ์เชิงประสิทธิภาพของตัวแบบบริการที่อัตราค่าโหลดแบบหลายอัตราที่ขนาดแคชร้อยละ 30 (แผนภาพที่ 3-8) แสดงตามลำดับประเภทแคชและประเภทบริการและ (แผนภาพที่ 9-11) แสดงตามลำดับประเภทบริการและชุดข้อมูล เพื่อแสดงสมรรถนะทางเทคนิค จากแผนภาพที่ 3-8 พบว่าอัตราประหยัดค่าโหลดข้อมูลของตัวแบบบริการ HDD Isolated มีอัตราประหยัดมากกว่า SSD Isolated และ RAM Isolated ที่ชุดข้อมูล BO และ NY ที่ขนาดแคชร้อยละ 30 แบบหลายอัตรา และกรณี Shared ตัวแบบบริการ HDD Shared มีอัตราประหยัดมากกว่า SSD Shared และ RAM Shared ที่ชุดข้อมูล BONY ที่ขนาดแคชร้อยละ 30 แบบหลายอัตรา, อัตราประหยัดเวลาโหลดข้อมูลของตัวแบบบริการ RAM Isolated มีอัตราประหยัดมากกว่า SSD Isolated และ HDD Isolated ที่ชุดข้อมูล BO และ NY ที่ขนาดแคชร้อยละ 30 แบบหลายอัตรา และกรณี Shared ตัวแบบบริการ RAM Shared มีอัตราประหยัดมากกว่า SSD Shared และ HDD Shared ที่ชุดข้อมูล BONY ที่ขนาดแคชร้อยละ 30 แบบหลายอัตรา, ตัวแบบบริการ HDD Isolated ที่ชุดข้อมูล BO มีอัตราประหยัดค่าโหลดข้อมูลเท่ากับร้อยละ 14.4166392235241 มากที่สุดเมื่อเปรียบเทียบกับกรณีที่เหลือที่ขนาดแคชร้อยละ 30 แบบหลายอัตรา, ตัวแบบบริการ RAM Shared มีอัตราประหยัดเวลาโหลดข้อมูลเท่ากับร้อยละ 20.3890570063422 มากที่สุดเมื่อเปรียบเทียบกับกรณีที่เหลือที่ขนาดแคชร้อยละ 30 แบบหลายอัตรา และจากแผนภาพที่ 9-11 พบว่าตัวแบบบริการแคชมีอัตราประหยัดค่าโหลดข้อมูลอยู่ในช่วง ร้อยละ 7.40 ถึง 14.41 ที่ขนาดแคชร้อยละ 30 แบบหลายอัตราและตัวแบบบริการแคชมีอัตราประหยัดเวลาโหลดข้อมูลอยู่ในช่วงร้อยละ 11.77 ถึง 20.38 ที่ขนาดแคชร้อยละ 30 แบบหลายอัตรา



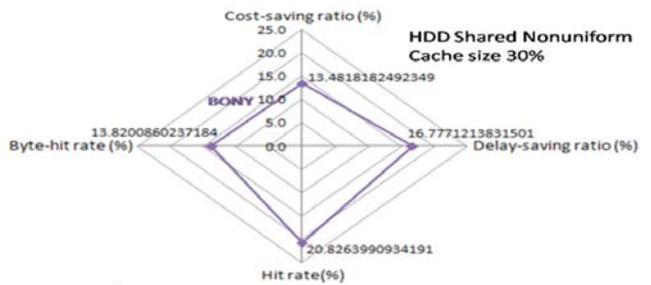
แผนภาพที่ 3 ผลการเปรียบเทียบของตัวแบบบริการ RAM Isolated



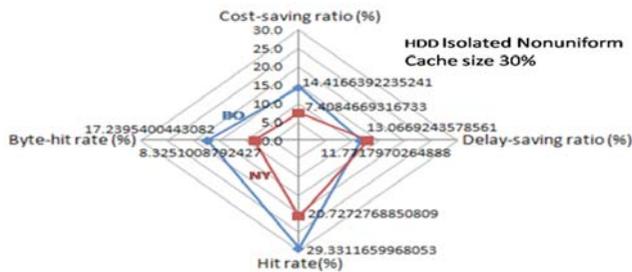
แผนภาพที่ 7 ผลการเปรียบเทียบของตัวแบบบริการ SSD Shared



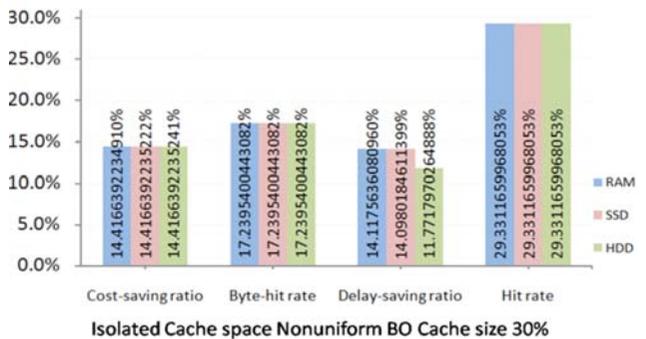
แผนภาพที่ 4 ผลการเปรียบเทียบของตัวแบบบริการ SSD Isolated



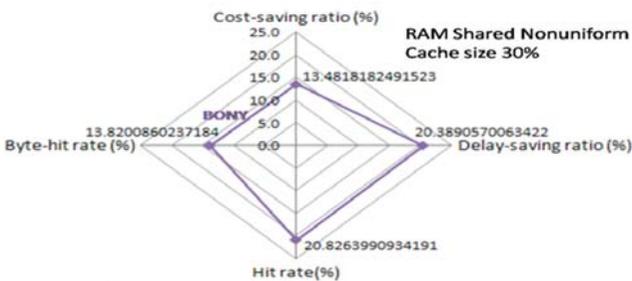
แผนภาพที่ 8 ผลการเปรียบเทียบของตัวแบบบริการ HDD Shared



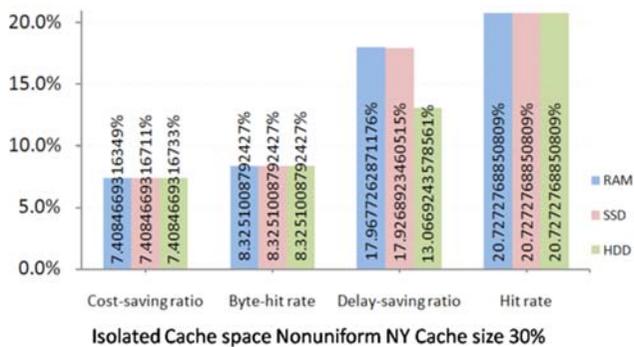
แผนภาพที่ 5 ผลการเปรียบเทียบของตัวแบบบริการ HDD Isolated



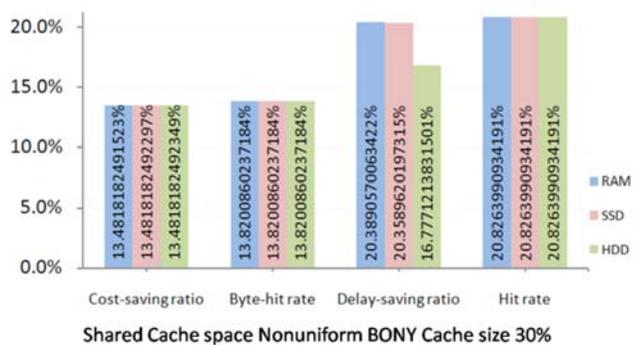
แผนภาพที่ 9 ผลการเปรียบเทียบเชิงประสิทธิภาพของตัวแบบบริการที่ประเภทบริการแบบ Isolated ที่ชุดข้อมูล BO



แผนภาพที่ 6 ผลการเปรียบเทียบของตัวแบบบริการ RAM Shared



แผนภาพที่ 10 ผลการเปรียบเทียบเชิงประสิทธิภาพของตัวแบบบริการ ที่ประเภทบริการแบบ Isolated ที่ชุดข้อมูล NY



แผนภาพที่ 11 ผลการเปรียบเทียบเชิงประสิทธิภาพของตัวแบบบริการ ที่ประเภทบริการแบบ Shared ที่ชุดข้อมูล BONY

สรุปและอภิปรายผล

งานวิจัยนี้นำเสนอตัวแบบบริการทางเทคนิคและทางเศรษฐศาสตร์สำหรับบริการแคชคลาวด์แบบใช้ร่วมกันที่ฝั่งไคลเอ็นท์ ผลการวิจัยแสดงให้เห็นว่าตัวแบบบริการทางเทคนิค HDD Isolated Cache space มีอัตราประหยัดค่าไหลดข้อมูลจากคลาวด์สูงสุดร้อยละ 14.41 เพราะราคาต่อหนึ่งหน่วยความจุต่ำกว่า RAM และ SSD และตัวแบบบริการทางเทคนิค RAM Shared Cache space มีอัตราประหยัดเวลาไหลดข้อมูลมากที่สุดเท่ากับร้อยละ 20.38 เนื่องจากใช้หน่วยเก็บข้อมูลความเร็วสูงเป็นพื้นที่แคช ส่วนตัวแบบทางเศรษฐศาสตร์แนะนำให้กำหนดราคาค่าใช้จ่ายบริการต่อเดือนโดยเฉลี่ย 16.34 ดอลลาร์สำหรับชุดข้อมูลที่ใช้ศึกษาซึ่งคำนวณมาจากค่าใช้จ่ายในการลงทุน CapEx และค่าใช้จ่ายใน

การดำเนินงาน OpEx ทั้ง 2 ส่วนคือค่าใช้จ่ายคงที่ OpExf และค่าใช้จ่ายผันแปร OpExv

ข้อเสนอแนะ

ข้อเสนอแนะเพิ่มเติมสำหรับตัวแบบทางเศรษฐศาสตร์ คือ การพิจารณาปรับตัวแปรในกลุ่มค่าใช้จ่ายในการลงทุน CapEx จากรูปแบบการเช่าพื้นที่โคโลเป็นรูปแบบของการเช่าบริการโครงสร้างพื้นฐานคลาวด์ เพื่อให้บริการแคช CaaS

เอกสารอ้างอิง

Ali, W., Shamsuddin, S. M., and Ismail, A. S. 2012. "Intelligent Naïve Bayes-based approaches for Web proxy caching." *Knowledge-Based Systems*, 31: 162-175.

Bakiras, S. and Loukopoulos, T. 2005. "Increasing the Performance of CDNS Using Replication and Caching: A Hybrid Approach." In *Parallel and Distributed Processing Symposium Proceedings*. 19th IEEE International, Denver, April 3-8, 2005 : 92.

Banditwattanawong, T., and Uthayopas, P. 2014. "A Client-Side Cloud Cache Replacement Policy." *ECTI Transactions on Computer and Information Technology: Special section on papers selected from ECTI-CON 2014*, 8(2): 113-121.

Banditwattanawong, T., Masdisornchote, M., and Uthayopas, P. 2016 "Multi-provider cloud computing network infrastructure optimization" *Future Generation Computer Systems*, 55: 116-128.

Chiu, D., and Agrawal, G. 2010. "Evaluating caching and storage options on the Amazon Web Services Cloud." *Proc. 11th ACM/IEEE Int. Conf. Grid Computing, IEEE Computer Society, Brussels, October 25-28, 2010*: 17-24.

Chiu, D., Shetty, A., and Agrawal, G. 2010. "Elastic cloud caches for accelerating service-oriented computations." In *High Performance Computing*,

- Networking, Storage and Analysis 2010 International Conference, New Orleans, November 13-19, 2010** : 1-11.
- Chockler, G., Laden, G., and Vigfusson, Y. 2010. "Data caching as a cloud service." **Proceedings of the 4th International Workshop on Large Scale Distributed Systems and Middleware. ACM, Zurich, July 25-28, 2010** : 18-21.
- Chockler, G., Laden, G., and Vigfusson, Y. 2011. "Design and implementation of caching services in the cloud." **IBM Journal of Research and Development**, 55(6), 9: 1-11.
- Dash, D., Kantere, V., and Ailamaki, A. 2009. "An economic model for self-tuned cloud caching" **IEEE 25th International Conference on Data Engineering, Shanghai, March 29 - April 2, 2009**: 1687-1693.
- ERC. (2011). "Study on Thailand Power Tariffs 2011-2015." Retrieved September 1, 2015, from http://www.erc.or.th/ERCWeb/Upload/.../เอกสารแนบ_อัตราค่าไฟฟ้า.pdf
- Feldman, A., Caceres, R., Douglis, F., Glass, G., and Rabinovich, M. 1999. "Performance of Web Proxy Caching in Heterogeneous Bandwidth Environments." **Proceedings of INFOCOM 99, New York, Mar 21-25, 1999** 1 : 107-116.
- Foong, K.Y., and Delcroix, J., C. 2015. "Market Trends: New Revenue Opportunities for Telecom Carriers in 2015" **Gartner**, Retrieved September 22, 2015, from <http://docslide.us/documents/gartner-market-trends-revenue-opportunities-for-telecom-carriers.html>
- Han, H., Lee, Y. C., Shin, W., Jung, H., Yeom, H. Y., and Zomaya, A. Y. 2012. "Cashing in on the Cache in the Cloud." **IEEE Trans. Parallel and Distributed Systems**, 23(8): 1387-1399.
- Goering, R. (2011). "ARM TechCon Paper: Why DRAM Latency is Getting Worse", Retrieved September 2, 2015, from http://community.cadence.com/cadence_blogs_8/b/ii/archive/2011/11/17/arm-techcon-paper-why-dram-latency-is-getting-worse
- Kantere, V., Dash, D., Francois, G., Kyriakopoulou, S., and Ailamaki, A. 2011. "Optimal Service Pricing for a Cloud Cache." **IEEE Transactions Knowledge and Data Engineering**, 23(9): 1345-1358.
- Kiani, S. L., Anjum, A., Munir, K., McClatchey, R., and Antonopoulos, N. 2011. "Towards Context Caches in the Clouds." **Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference, Victoria, December 5-8, 2011** : 403-408.
- KSC. (2015). "BIZ CO-LOCATION" Retrieved September 22, 2015, from <http://www.ksc.net/TH/Products-IDC-BizCoLocation.html>
- Liu, F., Makaroff, D., and Elnaffar, S. 2005. "Classifying e-commerce workloads under dynamic caching." **Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Waikoloa, October 10-12, 2005**, 3 : 2819-2824.
- Mach, W., and Schikuta, E. 2011. "A consumer-provider cloud cost model considering variable cost." **Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference, Sydney, December 12-14, 2011** : 628-635.
- Mell, P. and Grance, T. 2011. "The NIST Definition of Cloud Computing." Retrieved March 11, 2013, from the National Institute of Standards and Technology, from <http://csrc.nist.gov/groups/SNS/cloud-computing/>

NSF and NLANR. 2010. "IRCache." Retrieved June 2, 2010, from <http://www.ircache.net/>

Podlipnig, S. and Böszörményi, L. 2003. "A Survey of Web Cache Replacement Strategies." *ACM Computing Surveys*, 35(4) : 374-398.

Samsung. (2015). "Why SSDs Are Awesome", Retrieved September 7, 2015, from <http://www.samsung.com/global/business/semiconductor/minisite/SSD/global/html/whitepaper/whitepaper01.html>

Sriwiroj, C., and Banditwattanawong, T. 2015a, "An economic model for client-side cloud caching service.", In *Knowledge and Smart Technology (KST), 2015 7th International Conference on IEEE, Chonburi, January 28-31, 2015* : 131-136.

Sriwiroj, C., and Banditwattanawong, T. 2015b. "A cost model for client-side caas." *Information Science and Applications*, 339: 361-368.

Wang, J. 1999. "A survey of web caching schemes for the Internet" *ACM SIGCOMM Computer Communication Review*, 29(5): 36-46.

Xiao, L., Zhang, X., Andrzejak, A., and Chen, S. 2004. "Building a Large and Efficient Hybrid Peer-to-Peer Internet Caching System." *IEEE TKDE*, 16(6) : 754-769.



>> จตุรงค์ ศรีวีโรจน์

สำเร็จการศึกษา ปริญญาโท วิทยาศาสตร์มหาบัณฑิต สาขาเทคโนโลยีสารสนเทศทางธุรกิจ จากจุฬาลงกรณ์มหาวิทยาลัย และกำลังศึกษาในหลักสูตรปรัชญาดุษฎีบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม

ปัจจุบันทำงานตำแหน่งนักวิเคราะห์ ฝ่ายเทคโนโลยีสารสนเทศและการจัดการข้อมูล ศูนย์พันธุวิศวกรรมและเทคโนโลยีชีวภาพแห่งชาติ



>> ผู้ช่วยศาสตราจารย์ ดร.เทพฤทธิ์ บัณฑิตวัฒนาวงศ์

สำเร็จการศึกษา ปริญญาเอก (Ph.D.IT.) สาขา Informatics จาก The Graduate University for Advanced Studies ประเทศญี่ปุ่น ปริญญาโท (M.Eng.) สาขา Computer Science สถาบันเทคโนโลยีแห่งเอเชีย และปริญญาตรี (วศ.บ. เกียรตินิยม) สาขาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปัจจุบันดำรงตำแหน่งผู้ช่วยศาสตราจารย์ ประจำหลักสูตรปรัชญาดุษฎีบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม