



ใบรับรองวิทยานิพนธ์
บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)

ปริญญา

วิศวกรรมคอมพิวเตอร์

วิศวกรรมคอมพิวเตอร์

สาขา

ภาควิชา

เรื่อง การพัฒนาเทคนิคและอัลกอริทึมสำหรับบริหารทรัพยากรอย่างมีประสิทธิภาพบนระบบ
จัดเก็บข้อมูล แบบกลุ่มเมฆ

Development of Technique and Algorithm for Efficient Resource Management of Cloud
Storage System

นามผู้วิจัย นายกนกร จินดารักษ์

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(ผู้ช่วยศาสตราจารย์ ภูษงค์ อุทโยภาส, Ph.D.)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

(รองศาสตราจารย์ ประคนเดช นีละคุปต์, M.Eng.)

หัวหน้าภาควิชา

(ผู้ช่วยศาสตราจารย์ ภูษงค์ อุทโยภาส, Ph.D.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

(รองศาสตราจารย์กัญญา ชีระกุล, D.Agr.)

คณบดีบัณฑิตวิทยาลัย

วันที่ เดือน พ.ศ.

วิทยานิพนธ์

เรื่อง

การพัฒนาเทคนิคและอัลกอริทึมสำหรับบริหารทรัพยากรอย่างมีประสิทธิภาพ
บนระบบจัดเก็บข้อมูล แบบกลุ่มเมฆ

Development of Technique and Algorithm for Efficient Resource Management
of Cloud Storage System

โดย

นายกนธร จินดารักษ์

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์
เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)

พ.ศ. 2555

กนธร จินดารักษ์ 2555: การพัฒนาเทคนิคและอัลกอริทึมสำหรับบริหารทรัพยากรอย่างมีประสิทธิภาพบนระบบจัดเก็บข้อมูล แบบกลุ่มเมฆ ปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์) สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก: ผู้ช่วยศาสตราจารย์ภูงศค์ อุทโยภาศ, Ph.D. 38 หน้า

วิทยานิพนธ์นี้ เป็นการวิจัยและพัฒนาเทคนิคสำหรับการบริหารทรัพยากรบนระบบจัดเก็บข้อมูลแบบกลุ่มเมฆอย่างมีประสิทธิภาพ เพื่อผลลัพธ์ของการบริการที่ดีและเกิดความคุ้มค่าสูงสุด โดยคำนึงถึงทรัพยากรของระบบที่มีอยู่อย่างจำกัด งานวิจัยนี้มุ่งเน้นไปที่การเร่งประสิทธิภาพโดยการปรับสมดุลภาระงานของระบบด้วยการจัดวางวัตถุเก็บข้อมูลและการทำสำเนาวัตถุเก็บข้อมูลแบบพลวัต เพื่อกระจายภาระงานสำหรับวัตถุที่มีความต้องการอย่างมากเพียงชั่วคราว

ในงานวิจัยนี้ได้ใช้ขั้นตอนวิธีเชิงพันธุกรรมมาช่วยในการปรับสมดุลภาระงาน จากการทดสอบโดยการจำลองแบบพบว่าระบบมีสมรรถนะดีขึ้น เมื่อภาระงานการเข้าใช้ข้อมูลแต่ละหน่วยเพิ่มขึ้นมาก เครื่องแม่ข่ายแต่ละเครื่องอาจรับภาระงานมากเกินไป การทำสำเนาวัตถุเก็บข้อมูลแบบพลวัตจะให้ผลลัพธ์ที่เหมาะสมกว่าสำหรับระบบที่มีการเข้าใช้งานจากผู้ใช้งานจำนวนมาก

Kanatom Jindarak 2012: Development of Technique and Algorithm for Efficient Resource Management of Cloud Storage System. Master of Engineering (Computer Engineering), Major Field: Computer Engineering, Department of Computer Engineering. Thesis Advisor: Assistant Professor Putchong Uthayopas, Ph.D. 38 pages.

This thesis focuses on the development of techniques for an efficient resource management of the cloud storage systems. The goal is to provide a better service under limited system resources. This research focus on the use of effective workload balancing and a dynamic replication strategy. This will help balance the demand of object that is accessed heavily during a short period of time.

The genetic algorithm approach is employed to help balance the workload, the result is satisfactory. From the evaluation using simulation, the proposed method gives good results. Anyway, as the workload increased, the access can cause some server to be overloaded. The use of additional dynamic replication can be useful in enhancing the system performance.

Student's signature

Thesis Advisor's signature

สิงสิทธิ์ มตาวิทย์ เกษตรศาสตร์

กิตติกรรมประกาศ

ผู้วิจัยขอกราบขอบพระคุณ ผศ. ดร. กุชงค์ อุทโยภาส ประธานกรรมการที่ปรึกษาวิทยานิพนธ์ และ รศ. ดร. ประคนเดช นิละคุปต์ ที่ให้คำปรึกษาในการทำงานวิจัย รวมไปถึงการเรียน และการใช้ชีวิตในการทำงาน ตลอดจนการตรวจแก้ไขวิทยานิพนธ์จนกระทั่งเสร็จสมบูรณ์ และกราบขอบพระคุณผู้แทนบัณฑิตวิทยาลัย ที่ได้ให้ความกรุณาตรวจแก้ไขวิทยานิพนธ์ให้สมบูรณ์ยิ่งขึ้น

ขอกราบขอบพระคุณอาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่าน ที่ได้อบรมสั่งสอนและมอบความรู้อันเป็นประโยชน์อย่างยิ่งในการนำไปใช้ประโยชน์ต่อไป ขอขอบพระคุณเจ้าหน้าที่ภาควิชาวิศวกรรมคอมพิวเตอร์และเพื่อนๆที่ทุกคน ที่ให้ความช่วยเหลือและคำแนะนำ

ด้วยความดีหรือประโยชน์อันใดเนื่องจากวิทยานิพนธ์นี้ ขอมอบแต่คุณพ่อ คุณแม่ ผู้อบรม และให้กำลังใจผู้วิจัยมาตลอดในทุกเรื่อง

กนธร จินดารักษ์
สิงหาคม 2555

สารบัญ

หน้า

สารบัญ	(1)
สารบัญภาพ	(2)
คำอธิบายสัญลักษณ์และคำย่อ	(3)
คำนำ	1
วัตถุประสงค์	3
การตรวจเอกสาร	4
อุปกรณ์และวิธีการ	8
อุปกรณ์	8
วิธีการ	8
ผลและวิจารณ์	25
ผล	25
วิจารณ์	32
สรุปและข้อเสนอแนะ	34
สรุป	34
ข้อเสนอแนะ	35
เอกสารและสิ่งอ้างอิง	36
ประวัติการศึกษาและการทำงาน	38

สารบัญภาพ

ภาพที่		หน้า
1	ระบบเก็บข้อมูลแบบกลุ่มเมฆ	9
2	หน่วยการทำงานย่อยภายในเซิร์ฟเวอร์ควบคุม	9
3	ในมุมมองระดับโครงสร้างพื้นฐานระหว่างผู้ใช้บริการและเครื่องแม่ข่ายเก็บข้อมูล	11
4	ความสัมพันธ์ของวัตถุเก็บข้อมูล เซิร์ฟเวอร์เก็บข้อมูล และผู้ใช้บริการ	12
5	รูปแบบการแลกเปลี่ยนขึ้นทั้ง 4 รูปแบบ	15
6	จำลองสถานการณ์การทำงานของการสำเนาแบบพลวัต	18
7	การทำงานย่อยภายในเซิร์ฟเวอร์ควบคุมสำหรับการสำเนาแบบพลวัต	20
8	ตัวอย่างจำนวนการเข้าใช้งานไฟล์จากคล็อกไฟล์จาก mirror1.ku.ac.th	24
9	กราฟเปรียบเทียบค่าของฟังก์ชันค่าความเหมาะสมที่จำนวนวัตถุเก็บข้อมูลต่างๆ กัน	25
10	เวลาเฉลี่ยในการดาวน์โหลดไฟล์เมื่อใช้ขั้นตอนวิธีเชิงพันธุกรรมและแบบสุ่ม	27
11	เวลาเฉลี่ยในการดาวน์โหลดไฟล์เมื่อมีการเพิ่มจำนวนวัตถุเก็บข้อมูลและการใช้งาน	28
12	เปรียบเทียบค่าเบี่ยงเบนการกระจายภาระงานเมื่อปรับค่า L และให้ P คงที่	30
13	เปรียบเทียบค่าเฉลี่ยการใช้พื้นที่เก็บข้อมูลเมื่อปรับค่า L และให้ P คงที่	30
14	เปรียบเทียบค่าเบี่ยงเบนการกระจายภาระงานเมื่อปรับค่า P และให้ L คงที่	31
15	เปรียบเทียบค่าเฉลี่ยการใช้พื้นที่เก็บข้อมูลเมื่อปรับค่า P และให้ L คงที่	32

คำอธิบายสัญลักษณ์และคำย่อ

IaaS = Infrastructure as a Service

PaaS = Platform as a Service

SaaS = Software as a Service



การพัฒนาเทคนิคและอัลกอริทึมสำหรับบริหารทรัพยากรอย่างมีประสิทธิภาพบนระบบ
จัดเก็บข้อมูล แบบกลุ่มเมฆ

Development of Technique and Algorithm for Efficient Resource Management
of Cloud Storage System

คำนำ

ในปัจจุบันเทคโนโลยีการประมวลผลแบบกลุ่มเมฆ (cloud computing) ได้รับความนิยมอย่างแพร่หลาย เนื่องจากเครือข่ายอินเทอร์เน็ตมีความเร็วที่สูงขึ้น ทำให้ผู้ใช้บริการสามารถเข้าถึงบริการต่างๆ ได้จากเครื่องคอมพิวเตอร์หรืออุปกรณ์พกพาได้อย่างสะดวก การบริการพื้นที่จัดเก็บข้อมูลแบบกลุ่มเมฆ (cloud storage) ก็เป็นอีกรูปแบบบริการหนึ่งที่กำลังได้รับความนิยมเนื่องจากพฤติกรรมการใช้งานอุปกรณ์คอมพิวเตอร์ได้เปลี่ยนแปลงไปดังที่ได้กล่าวมาแล้ว ระบบเก็บข้อมูลแบบกลุ่มเมฆในปัจจุบันมีผู้ให้บริการอยู่เป็นจำนวนมาก ซึ่งรูปแบบการบริการส่วนใหญ่จะให้บริการพื้นที่เก็บข้อมูลสำหรับผู้ใช้แต่ละคน โดยผู้ให้บริการจะพัฒนาแอปพลิเคชันเพื่อรองรับการให้บริการบนแพลตฟอร์มต่างๆ เพื่อตอบสนองความต้องการของผู้ใช้ ทั้งในคอมพิวเตอร์ส่วนบุคคลบนระบบปฏิบัติการที่แตกต่างกัน หรือในอุปกรณ์พกพาต่างๆ

คุณสมบัติที่สำคัญอย่างหนึ่งของระบบจัดเก็บข้อมูลแบบกลุ่มเมฆ คือการรองรับการใช้งานจากผู้ใช้งานจำนวนมากได้อย่างมีประสิทธิภาพ ซึ่งก็อาจจะใช้วิธีการเพิ่มหน่วยเก็บข้อมูล เพื่อเพิ่มทรัพยากรสำหรับรองรับการให้บริการจากผู้ใช้งาน แต่อีกวิธีการหนึ่งที่สามารถจัดสรรทรัพยากรของหน่วยเก็บข้อมูลได้อย่างคุ้มค่าคือการปรับสมดุลภาระงาน (load balancing) ของหน่วยเก็บข้อมูล

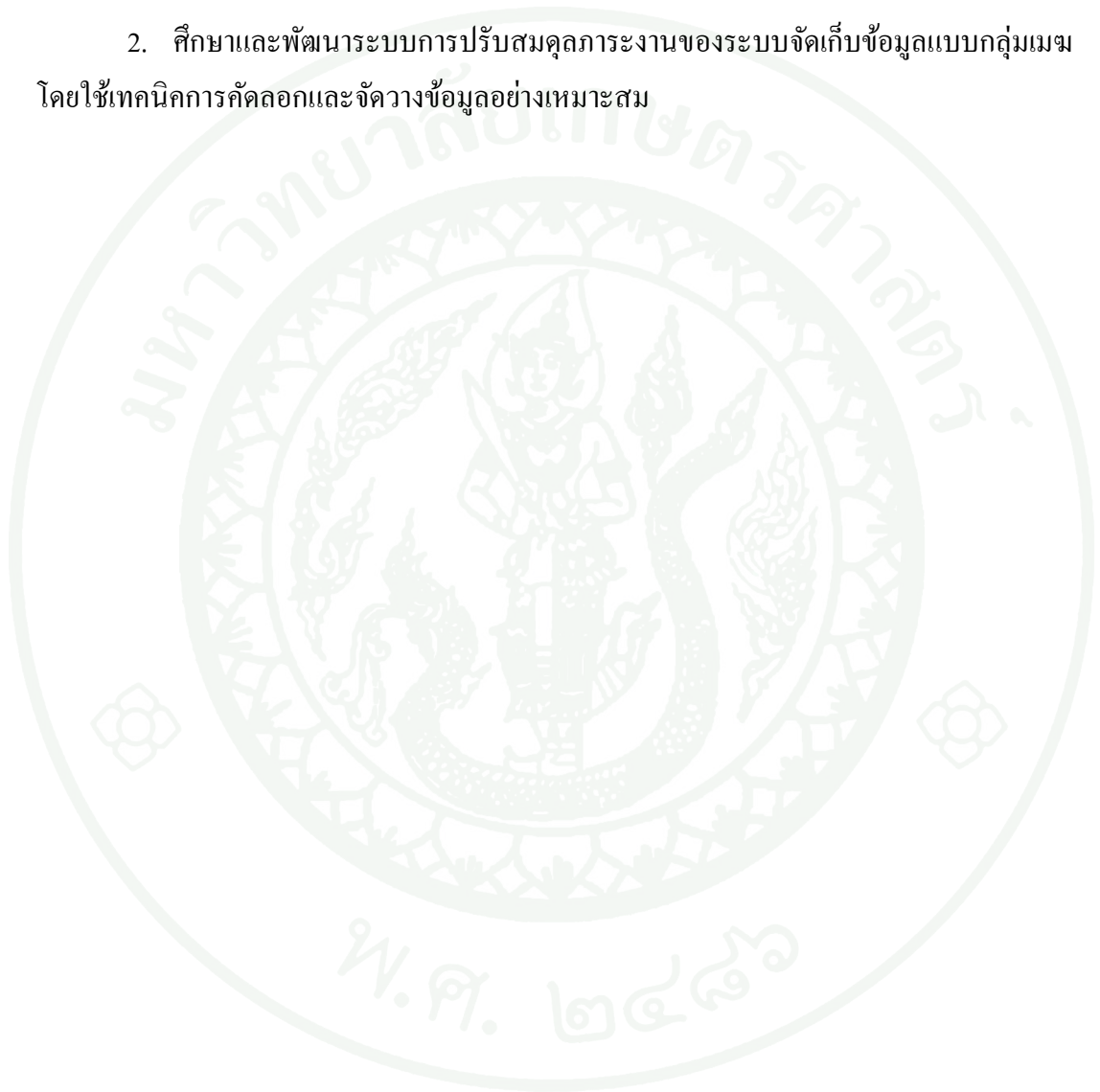
การปรับสมดุลภาระงานจากการเข้าใช้งานของผู้ใช้สามารถทำได้หลายวิธี แต่ถ้าพิจารณาในมุมมองของระบบการบริหารจัดการก็จะแบ่งได้เป็นสองรูปแบบคือ การจัดการแบบรวมศูนย์ (centralized system) และการจัดการแบบกระจาย (distributed system) ซึ่งจะมีข้อดีข้อเสียแตกต่างกัน สำหรับการปรับสมดุลภาระงานแบบรวมศูนย์นั้นจะสามารถดูแลจัดการระบบได้ง่ายกว่า เนื่องจากระบบจัดการทุกอย่างอยู่ที่ศูนย์กลาง แต่ข้อเสียก็คือการที่ภาระการจัดการทั้งหมดมารวมอยู่ที่ศูนย์กลางทำให้เซิร์ฟเวอร์ที่มีหน้าที่ควบคุมการทำงานการปรับสมดุลภาระงานต้องทำงานหนักในกรณีที่มีผู้ใช้บริการจำนวนมาก และหากเซิร์ฟเวอร์ดังกล่าวมีการทำงานผิดพลาดก็จะส่งผลกระทบต่อ

กับระบบทั้งหมด ส่วนการปรับสมดุลภาระงานแบบกระจายนั้นจะมีข้อดีคือเสถียรภาพของระบบจะสูงกว่า เนื่องจากภาระงานไม่ได้ขึ้นอยู่กับเซิร์ฟเวอร์เครื่องใดเครื่องหนึ่ง แต่การพัฒนาระบบก็จะซับซ้อนกว่า

งานวิจัยนี้ศึกษาเพื่อเสนออัลกอริทึมการปรับสมดุลภาระงานที่เหมาะสมกับระบบจัดเก็บข้อมูลแบบกลุ่มเมฆ หากการนำอัลกอริทึมแบบกระจายที่เหมาะสมมาประยุกต์ใช้กับระบบปรับสมดุลภาระงานในระบบจัดเก็บข้อมูลแบบกลุ่มเมฆ จะสามารถสร้างระบบจัดเก็บข้อมูลแบบกลุ่มเมฆที่สามารถรองรับภาระงานที่เกิดจากการใช้บริการของผู้ใช้ได้อย่างมีประสิทธิภาพและใช้ทรัพยากรที่มีอยู่ได้อย่างคุ้มค่ามากขึ้น

วัตถุประสงค์

1. ศึกษาและพัฒนาระบบการปรับสมดุลภาระงานของระบบจัดเก็บข้อมูลแบบกลุ่มเมฆ โดยการจัดวางข้อมูลอย่างเหมาะสม
2. ศึกษาและพัฒนาระบบการปรับสมดุลภาระงานของระบบจัดเก็บข้อมูลแบบกลุ่มเมฆ โดยใช้เทคนิคการคัดลอกและจัดวางข้อมูลอย่างเหมาะสม



การตรวจเอกสาร

ในหัวข้อนี้จะกล่าวถึงความรู้พื้นฐานของเทคโนโลยีการประมวลผลแบบกลุ่มเมฆ และงานที่เกี่ยวข้องกับงานวิจัยของเทคโนโลยีระบบเก็บข้อมูลแบบกลุ่มเมฆ รวมถึงอัลกอริทึมที่จะนำมาประยุกต์ใช้กับงานวิจัยนี้ด้วย

เทคโนโลยีการประมวลผลแบบกลุ่มเมฆเป็นแบ่งประเภทในมุมมองของการให้บริการ และรูปแบบการใช้งาน (National Institute of Standards and Technology [NIST], 2011) โดยการแบ่งตามมุมมองการให้บริการแบ่งได้ดังนี้ การให้บริการโครงสร้างพื้นฐาน (Infrastructure as a Service หรือ IaaS) เป็นการให้บริการโครงสร้างพื้นฐานของระบบคอมพิวเตอร์ ในมุมมองของผู้ใช้บริการจะเสมือนว่าเป็นการเช่าใช้บริการฮาร์ดแวร์ของระบบคอมพิวเตอร์จากผู้ให้บริการ เช่น การเช่าเครื่องคอมพิวเตอร์ การเช่าพื้นที่ให้บริการข้อมูล หรือ การให้บริการครบวงจรของ Amazon EC2 เป็นต้น ทั้งนี้ในระบบภายในของผู้ให้บริการอาจจะเป็นระบบเสมือนที่ทำให้ผู้ใช้บริการสามารถใช้บริการระบบได้เหมือนเป็นผู้ครอบครองฮาร์ดแวร์นั้นจริง โดยส่วนใหญ่แล้วการให้บริการโครงสร้างพื้นฐานจะคิดค่าใช้จ่ายจากการใช้งานจริงของผู้ใช้ เช่น การเช่าพื้นที่เก็บข้อมูลก็จะคิดตามขนาดความจุตามที่คุณต้องการ เป็นต้น นอกจากนี้จุดเด่นของระบบประมวลผลแบบกลุ่มเมฆก็คือ การที่ผู้ใช้บริการสามารถเพิ่มความต้องการในการใช้บริการได้เมื่อจะเป็น ทำให้ไม่สิ้นเปลืองค่าใช้จ่ายในการลงทุนที่เกินความจำเป็น

การให้บริการแพลตฟอร์ม (Platform as a Service หรือ PaaS) เป็นการให้บริการแพลตฟอร์มหรือสภาพแวดล้อมที่เอื้อต่อการพัฒนา หรือการเรียกใช้งานบริการเฉพาะอย่างให้ผู้ให้บริการสามารถเรียกใช้งานหรือเข้าไปใช้งานได้ อาทิเช่น การให้บริการสภาพแวดล้อมในการพัฒนาโปรแกรม หรือเว็บแอปพลิเคชันของ Google Apps Engine ที่มีพื้นฐานข้อมูล เว็บเซิร์ฟเวอร์ และเครื่องมือในการพัฒนา บริการให้กับนักพัฒนา เป็นต้น

การให้บริการซอฟต์แวร์ (Software as a Service หรือ SaaS) เป็นบริการที่เปิดให้ผู้ใช้งานสามารถเข้ามาใช้งานซอฟต์แวร์หรือเว็บแอปพลิเคชันที่ทางผู้ให้บริการเตรียมไว้ให้ได้ โดยส่วนใหญ่จะคิดค่าใช้จ่ายตามปริมาณการใช้งาน อาทิเช่น Gmail และ Picasa เป็นต้น

นอกจากนี้เทคโนโลยีการประมวลผลแบบกลุ่มเมฆยังสามารถแบ่งตามรูปแบบการใช้งาน (deployment model) ได้เป็น 4 ประเภทคือ ระบบประมวลผลแบบกลุ่มเมฆส่วนบุคคล (private cloud)

ระบบประมวลผลแบบกลุ่มเมฆชุมชน (community cloud) ระบบประมวลผลแบบกลุ่มเมฆแบบลูกผสม (hybrid cloud) และระบบประมวลผลแบบกลุ่มเมฆสาธารณะ (public cloud) ซึ่งการแบ่งกลุ่มตามรูปแบบการใช้งานดังกล่าวนี้ จะใช้มุมมองสิทธิ์ในการให้บริการเป็นหลัก เช่นระบบประมวลผลแบบกลุ่มเมฆส่วนบุคคลก็จะใช้งานได้เฉพาะภายในองค์กรเท่านั้น หรือระบบประมวลผลแบบกลุ่มเมฆสาธารณะก็จะสามารถมีผู้ใช้งานได้โดยไม่จำกัดว่าจะเป็นผู้ใช้งานหรือนอกองค์กร

ในส่วนของระบบเก็บข้อมูลแบบกลุ่มเมฆ (cloud Storage) นั้นปัจจุบันก็กำลังเป็นที่นิยมอย่างแพร่หลาย เนื่องจากผู้ใช้งานในปัจจุบันส่วนใหญ่จะมีอุปกรณ์คอมพิวเตอร์หรืออุปกรณ์พกพามากกว่าในอดีต การทำงานหรือการเก็บข้อมูลก็อาจจะกระจายอยู่ในอุปกรณ์ต่างๆ ทำให้ไม่สะดวกเมื่อจะใช้งานข้อมูลหรือไฟล์เหล่านั้น จึงทำให้เกิดบริการระบบเก็บข้อมูลแบบกลุ่มเมฆ ที่ผู้ให้บริการไม่ต้องสนใจว่าจะเก็บไฟล์ไว้ที่ใด แต่สามารถเข้าถึงไฟล์ที่ต้องการได้จากทุกๆ ที่มีระบบอินเทอร์เน็ตและสามารถเชื่อมต่อกับผู้ให้บริการได้ ซึ่งการให้บริการระบบเก็บข้อมูลแบบกลุ่มเมฆ นั้นมีทั้งแบบการให้บริการซอฟต์แวร์ และแบบการให้บริการโครงสร้างพื้นฐาน อาทิเช่น บริการของ DropBox ซึ่งเป็นการให้บริการแบบการให้บริการซอฟต์แวร์ โดยให้ผู้ให้บริการติดตั้งซอฟต์แวร์ที่ให้ดาวน์โหลดมาใช้บริการฟรี เมื่อมีการบันทึกไฟล์ไว้ในเครื่องของผู้ใช้ในแฟ้มข้อมูลที่กำหนด ซอฟต์แวร์ที่ติดตั้งไว้ก็จะทำการซิงโครไนซ์ไฟล์กับฝั่งเซิร์ฟเวอร์ ซึ่งทางผู้ให้บริการจะมีซอฟต์แวร์ไว้ให้ผู้ให้บริการทุกแพลตฟอร์มทั้งอุปกรณ์พกพารุ่นต่างๆ และระบบปฏิบัติการหลายระบบที่เป็นที่นิยม หรือบริการระบบจัดเก็บข้อมูลแบบกลุ่มเมฆที่ให้บริการแบบโครงสร้างพื้นฐานของ Nomadesk ซึ่งจะมีความแตกต่างจากการให้บริการแบบซอฟต์แวร์ของ DropBox ตรงที่เมื่อผู้ใช้ติดตั้งซอฟต์แวร์ที่กำหนดแล้ว ผู้ให้บริการจะมองเห็น ส่วนเก็บข้อมูลเป็นเสมือนมีดิสก์อยู่ที่เครื่องคอมพิวเตอร์ของผู้ให้บริการ ซึ่งก็แล้วแต่ความต้องการและวัตถุประสงค์ของผู้ใช้บริการว่าต้องการใช้งานในลักษณะใด

ในปัจจุบันระบบเก็บข้อมูลแบบกลุ่มเมฆเพิ่งจะเริ่มได้รับความนิยม ซึ่งอาจจะเนื่องมาจากความเร็วและประสิทธิภาพของเครือข่ายอินเทอร์เน็ตที่เพียงพอกับความต้องการในการส่งข้อมูลระหว่างกัน ทำให้งานวิจัยที่เกี่ยวข้องยังอาจจะไม่ครอบคลุมในด้านต่างๆ ครบถ้วนเหมือนงานวิจัยในด้านอื่นๆ อย่างไรก็ตามก็ยังมีงานวิจัยที่เกี่ยวข้องอยู่บ้าง ดังนี้

เนื่องจากการให้บริการของระบบเก็บข้อมูลแบบกลุ่มเมฆ จะเป็นการให้บริการผู้ใช้เป็นจำนวนมาก จึงมีนักวิจัยที่วิเคราะห์โครงสร้างของระบบการบริหารจัดการสำหรับระบบเก็บข้อมูลแบบกลุ่มเมฆ (Ying and Yong, 2009) โดยระบบเก็บข้อมูลแบบกลุ่มเมฆในปัจจุบันส่วนใหญ่โครงสร้าง

ภายในจะประกอบด้วย โหนดเก็บข้อมูล ซึ่งในระบบเก็บข้อมูลแบบกลุ่มเมฆ จะมีโหนดเก็บข้อมูลเป็นจำนวนมาก โดยจะมีระบบการจัดการของโหนดเก็บข้อมูล ที่คอยดูแลจัดการ โหนดเก็บข้อมูลทั้งหมด และยังมีส่วนที่เป็นส่วนติดต่อกับผู้ใช้หรือเป็นส่วนตัวกลางที่เชื่อมต่อการให้บริการ เรียกว่า Application Interface Layer ในโหนดเก็บข้อมูล นั้นจะประกอบไปด้วย Storage Device, Data Pooling และ Data Pooling Management Layer โดย Storage Node แต่ละ Node จะเชื่อมต่อกันด้วยระบบ Network

ใน Data Pooling จะประกอบด้วย Online Data Pooling, Backup Pooling, History Data Pooling, Data Warehouse และ Data Table Model โดยหลักๆนั้น ใน Data Pooling จะทำหน้าที่ให้บริการข้อมูลกับผู้ใช้ตามคำร้องขอ ซึ่งจะเก็บข้อมูลใหม่ๆ ไว้ใน Online Data Pooling ซึ่งจะใช้หน่วยความจำหลักเป็นที่เก็บ ส่วนข้อมูลเก่ากว่านั้นจะเก็บไว้ใน History Data หากมีความผิดพลาดข้อมูลในหน่วยความจำหลักหาย ซึ่งอาจจะเกิดได้จากหลายสาเหตุก็จะสามารถดึงข้อมูลใน Backup Pooling ได้ทันที นอกจากการให้บริการข้อมูลดังกล่าวมาแล้ว Data Pooling ยังทำหน้าที่ในการสำรองข้อมูลอีกด้วย โดยจะทำการ Snapshot ไว้ใน Data Warehouse

ระบบการเข้าถึงไฟล์ในระบบเก็บข้อมูลแบบกลุ่มเมฆในปัจจุบันที่มีอยู่หลายระบบ อาทิ เช่น GFS ของกูเกิ้ล Blue Cloud ของ IBM และ Elastic Cloud ของ Amazon ซึ่งการทำงานในการเข้าถึงไฟล์จากการร้องขอของผู้ใช้นั้นจะต้องร้องขอไปที่ เครื่อง Central Index Server กลาง จากนั้น เครื่อง Central Index Server ก็จะทำการค้นหาว่าไฟล์ที่ผู้ใช้ร้องขอนั้นอยู่ที่ Chunk Server เครื่องใด และตอบกลับไปยังผู้ใช้ แต่ลักษณะการทำงานดังกล่าวมา หากมีการร้องขอจากผู้ใช้เป็นจำนวนมาก และอาจจะมีคำร้องขอมาพร้อมๆกัน ก็จะเกิดปัญหาคอขวดที่ Central Index Server เนื่องจากเป็นผู้รับภาระในการหา Chunk Server ตามคำร้องขอทั้งหมด จึงมีการนำเสนอแนวทางใหม่คือไม่มี Central Index Server แต่จะแทนที่ด้วย Gateway ธรรมดาๆแทน โดย Gateway จะส่ง request ให้กับ Chuck Server ที่มี Index อยู่ในตัวเอง จากนั้น Chuck Server จะทำการค้นหาตำแหน่งของไฟล์ที่ต้องการในกลุ่ม Chuck Server ด้วยกันด้วย P2P Protocol จากนั้น Chuck Server ใดที่ทราบว่าไฟล์นี้อยู่ที่ตนเองก็จะตอบกลับไปยัง Client ที่ร้องขอมา (Xu *et al.*, 2009)

จากลักษณะการทำงานที่ได้กล่าวมา หลังจากได้ทดลองกับระบบต้นแบบแล้ว ผลลัพธ์ที่ได้พบว่าแม้จะเพิ่มจำนวน Chunk Server ขึ้น แต่ก็ไม่มีแนวโน้มว่าการเข้าถึงไฟล์จากการร้องขอจะช้าลง ซึ่งจะแตกต่างกับระบบที่มี Central Index Server ที่ยิ่งเพิ่ม Chunk Server ก็จะทำให้การค้นหาที่

Central Index Server ทำให้ช้าลง นอกจากนี้ทางผู้จัดทำยังได้กล่าวถึงการพัฒนาต่อในอนาคตว่า อาจเพิ่มความเร็วได้การอ่านและเขียนข้อมูลแบบไปป์ไลน์อีกด้วย

ยังมีงานวิจัยที่พัฒนาปรับปรุงรูปแบบการเข้าถึงไฟล์ให้มีประสิทธิภาพยิ่งขึ้นไปอีก โดยการใช้ Cloud Ontology มาทดแทน Central Index Server โดยวิธีการเข้าถึงไฟล์หรือข้อมูลนั้นยังมีขั้นตอนแบบเดิมดังที่ได้อธิบายแล้วในงานวิจัยก่อนหน้านี้ แต่สิ่งที่เปลี่ยนแปลงไปคือ ขั้นตอนการประมวลผลเพื่อหา Chunk Server ที่เก็บข้อมูลที่อยู่ของขอต้องการนั้น จะทำการค้นหาด้วย Cloud Ontology ซึ่งในงานวิจัยกล่าววาระบบจะเข้าใจความหมายของข้อมูลในเชิงลึกได้มากกว่าการค้นหาแบบเดิมๆ จึงทำให้การค้นหาทำได้เร็วกว่า (Feel and Khafagy, 2011)

งานวิจัยที่ทำเกี่ยวกับเรื่องการจัดวางวัตถุเก็บข้อมูลเพื่อพัฒนาประสิทธิภาพของระบบเก็บข้อมูลแบบกลุ่มเมฆก็มีอยู่หลายงานวิจัยซึ่งจะมีรายละเอียดปลีกย่อยแตกต่างกันไป อาทิเช่นการปรับสมดุลภาระงาน โดยการบริหารและรักษาสมดุลของการร้องขอเพื่อเข้าถึงข้อมูล (Zeng and Veeravalli, 2008) ในบางงานวิจัยก็ใช้ทั้งการจัดวางวัตถุเก็บข้อมูลและการทำสำเนาข้อมูลเพื่อสร้างสมดุลภาระงานในระบบซึ่งก็จะมีวิธีที่แตกต่างกันไป อาทิเช่น ใช้พื้นที่ที่เหลือในการเก็บข้อมูลและอัตราการคงอยู่โหนดเก็บข้อมูลเป็นเกณฑ์ในการพิจารณาเพื่อจัดวางวัตถุเก็บข้อมูลที่โดนทำสำเนา (Myint and Naing, 2011) และยังมีงานวิจัยที่สร้างสมดุลภาระงานแบบพลวัต โดยการพิจารณาจากภาระงานของแต่ละวัตถุเก็บข้อมูล และจะพิจารณาทำการสำเนาข้อมูลเมื่อถึงช่วงเวลาที่กำหนด (Qingsong *et al.*, 2010)

เนื่องจากระบบการประมวลผลแบบกลุ่มเมฆนั้นเป็นระบบที่มีขนาดใหญ่ ปัญหาการรักษาสมดุลภาระงานในระบบจึงเป็นปัญหาที่มีขนาดใหญ่ตามด้วย จึงมีงานวิจัยที่ทดลองใช้ขั้นตอนวิธีเชิงพันธุกรรมมาแก้ปัญหาดังกล่าว อาทิเช่น การใช้ขั้นตอนวิธีเชิงพันธุกรรมในการจัดการภาระงานที่เกิดจากโปรแกรมประยุกต์ให้มีความสมดุลกับระบบปฏิบัติการเสมือนในระบบ (Zhu *et al.*, 2011) หรืออีกงานวิจัยจะใช้ขั้นตอนวิธีเชิงพันธุกรรมในการจัดวางโปรเซสในเซิร์ฟเวอร์ที่เหมาะสมเพื่อกระจายภาระงานในระบบ (Yusoh and Maolin, 2010)

อุปกรณ์และวิธีการ

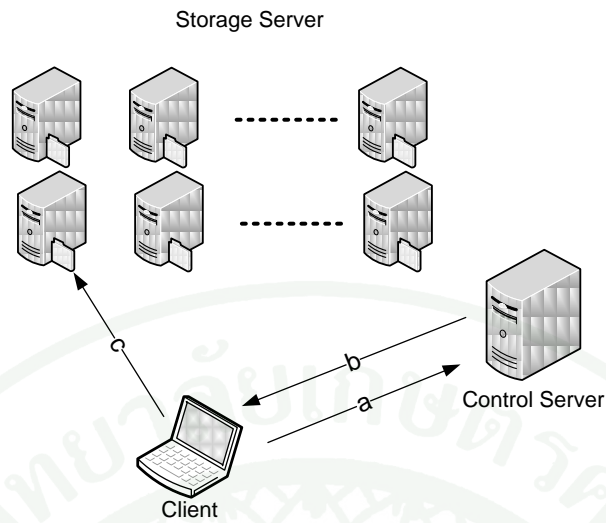
อุปกรณ์

1. เครื่องคอมพิวเตอร์
 - 1.1 หน่วยประมวลผล Intel Core 2 Duo, 2.26 GHz
 - 1.2 หน่วยความจำ 8 GB DDR3
2. ระบบปฏิบัติการ Mac OS X รุ่น 10.6.8
3. ตัวแปลภาษาจาวา (Java SE Runtime Environment) รุ่น 1.6.0

วิธีการ

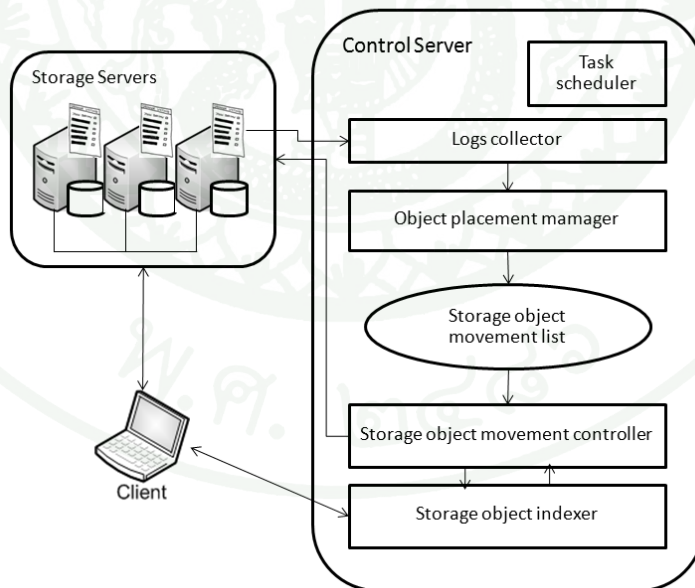
ในงานวิจัยนี้จะออกแบบให้ระบบจัดเก็บข้อมูลแบบกลุ่มเมฆมีเครื่องเซิร์ฟเวอร์หลักเพื่อเก็บ meta data ของวัตถุเก็บข้อมูล (storage object) ทั้งหมดที่ถูกเก็บไว้ที่เซิร์ฟเวอร์เครื่องใดในระบบ เมื่อมีการร้องขอใช้งานวัตถุเก็บข้อมูล เซิร์ฟเวอร์หลักก็จะแจ้งกลับไปยังเครื่องที่ร้องขอโดยระบุเครื่องเซิร์ฟเวอร์เก็บข้อมูล (storage server) ที่เก็บวัตถุเก็บข้อมูลที่ร้องขอไว้ จากนั้นเครื่องที่ร้องขอก็สามารถทำการเชื่อมต่อโดยตรงไปยังเซิร์ฟเวอร์ดังกล่าว จากภาพที่ 1 จะแสดงขั้นตอนการทำงานเมื่อมีการร้องขอจากเครื่องลูกข่ายเพื่อใช้งานวัตถุเก็บข้อมูล ซึ่งจะมีขั้นตอนจาก a ไป b และ c ตามลำดับ

จากโครงสร้างดังกล่าวการปรับสมดุลภาระงานจะต้องประมวลผลที่เครื่องเซิร์ฟเวอร์ควบคุม (control server) เพื่อบอกว่าการจัดวางวัตถุเก็บข้อมูล ที่กระจายตัวอยู่ในเซิร์ฟเวอร์เก็บข้อมูลในระบบควรอยู่ในรูปแบบใด จากนั้นจึงส่งการไปยังเซิร์ฟเวอร์เก็บข้อมูลในระบบ เพื่อปรับย้ายวัตถุเก็บข้อมูลตามความเหมาะสม



ภาพที่ 1 ระบบเก็บข้อมูลแบบกลุ่มเมฆ

สำหรับโครงสร้างภายในของระบบ นั้นได้ออกแบบไว้โดยแบ่งออกเป็น 5 ส่วนการทำงาน
 ย่อยดังแสดงไว้ดังแสดงในภาพที่ 2



ภาพที่ 2 หน่วยการทำงานย่อยภายในเซิร์ฟเวอร์ควบคุม

โดยแต่ละส่วนการทำงานจะมีหน้าที่ต่าง ๆ กันดังนี้

1. Logs Collector ทำหน้าที่เก็บ Log การเข้าใช้งาน ไฟล์หรือวัตถุเก็บข้อมูล จาก เซิร์ฟเวอร์เก็บข้อมูล โดยในการทำงานจะคอยเก็บ Log ตามเวลาที่ Task Scheduler ได้กำหนดไว้ โดย Log ที่ได้มานำจะเตรียมไว้สำหรับให้ Object Placement Manager ใช้งานต่อไป
2. Object Placement Manager เป็นส่วนที่ในงานวิจัยนี้จะให้ความสำคัญเป็นอย่างมาก เนื่องจากทำหน้าที่ในการนำ Log การเข้าใช้งาน ไฟล์หรือวัตถุเก็บข้อมูลที่ได้จาก Logs Collector มาใช้ในการวิเคราะห์ เพื่อหารูปแบบในการจัดวางวัตถุเก็บข้อมูล ในโหนดเก็บข้อมูลที่เหมาะสม เพื่อให้เกิดประสิทธิภาพสูงสุดในการกระจายภาระงานภายในระบบ โดยผลลัพธ์ที่ได้ก็คือข้อมูลที่ใช้ในการจัดวางวัตถุเก็บข้อมูลในรูปแบบใหม่
3. Storage Object Movement Controller ในส่วนการทำงานนี้ Storage Object Movement Controller จะนำข้อมูล Storage Object Movement List ที่ได้จาก Object Placement Manager มาทำการย้ายวัตถุเก็บข้อมูล ให้ไปอยู่ในโหนดเก็บข้อมูลที่กำหนด โดยจะทำการอ่านข้อมูลจาก storage object indexer เพื่อตรวจสอบว่าปัจจุบันวัตถุเก็บข้อมูลที่ต้องการย้ายอยู่ใน โหนดเก็บข้อมูลใด และหลังจากทำงานย้ายเสร็จแล้ว ก็จะทำการอัปเดตตำแหน่งใหม่ของวัตถุเก็บข้อมูลไปยัง storage object indexer
4. Storage Object Indexer ส่วนนี้จะป็นหน่วยการทำงานที่คอยดูแลข้อมูลความสัมพันธ์ระหว่างวัตถุเก็บข้อมูล และ โหนดเก็บข้อมูล โดยจะเก็บข้อมูลความสัมพันธ์ไว้เพื่อให้ส่วนการทำงานอื่นมาเรียกใช้งาน หากมีการเปลี่ยนแปลงความสัมพันธ์ดังกล่าว storage object indexer ก็จะทำการปรับปรุงข้อมูลให้ทันสมัยอยู่เสมอ
5. Task Scheduler จะทำหน้าที่หาเวลาที่เหมาะสมเพื่อร้องขอให้มีการปรับปรุงการกระจายภาระงาน รวมถึงการร้องขอให้มีการเก็บข้อมูลการเข้าถึงวัตถุเก็บข้อมูลด้วย

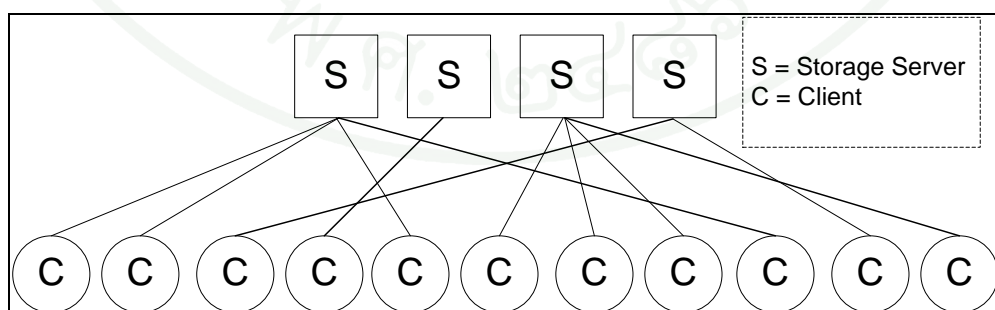
สำหรับแบบจำลองของโมเดลระบบจัดเก็บข้อมูลแบบกลุ่มเมฆ และวิธีการที่จะใช้ในการเพิ่มประสิทธิภาพของระบบโดยการกระจายภาระงานที่จะนำเสนอในงานวิจัยนี้มีดังนี้

1. โมเดลของระบบจัดเก็บข้อมูลแบบกลุ่มเมฆ

จากโครงสร้างของระบบจัดเก็บข้อมูลแบบกลุ่มเมฆดังที่กล่าวมาเป็นการอธิบายโครงสร้างในรูปแบบของโครงสร้างพื้นฐาน ในหัวข้อนี้จะกล่าวถึงแบบจำลองของระบบจัดเก็บข้อมูลแบบกลุ่มเมฆที่ใช้ในงานวิจัยนี้

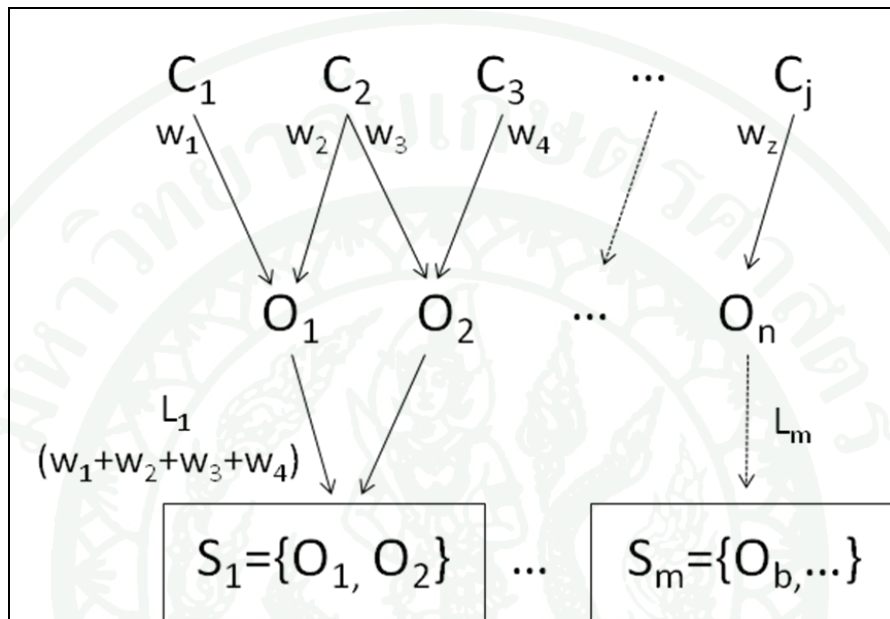
จากระบบข้างต้นที่กล่าวมา การปรับสมดุลภาระงาน (workload) คือการจัดวางตำแหน่งของวัตถุเก็บข้อมูลไปยังเซิร์ฟเวอร์เก็บข้อมูลที่เหมาะสม โดยในงานวิจัยนี้จะสมมุติให้เซิร์ฟเวอร์เก็บข้อมูลทุกเครื่องมีความสามารถในการรับภาระงานได้เท่ากัน โดยมีวัตถุเก็บข้อมูลที่สร้างภาระงานขนาดต่างๆกันในระบบ โดยการร้องขอใช้งานวัตถุเก็บข้อมูลจะเกิดจากผู้ใช้ (user) ซึ่งภาระงานในงานวิจัยนี้ได้กำหนดให้ภาระงานคือจำนวนแบนด์วิธที่เกิดขึ้นจริงจากการดาวน์โหลดวัตถุเก็บข้อมูลตัวหนึ่งตามช่วงระยะเวลาที่กำหนด ดังนั้นถึงแม้ว่าวัตถุเก็บข้อมูลจะมีการเข้าใช้งานหรือการดาวน์โหลดมากก็ไม่ได้หมายความว่า จะมีภาระงานที่มาก เพราะจะต้องดูแบนด์วิธที่ถูกดาวน์โหลดจากผู้ใช้งานด้วย

จากภาพที่ 3 จะสังเกตเห็นได้ว่าเซิร์ฟเวอร์เก็บข้อมูล (S) แต่ละเครื่องจะมีการเข้าใช้บริการของผู้ใช้บริการ (C) ไม่เท่าเทียมกัน แต่ก็ไม่ได้หมายความว่าเซิร์ฟเวอร์เก็บข้อมูลแต่ละเครื่องจะมีภาระงานที่แตกต่างกัน ทั้งนี้ก็ขึ้นอยู่กับภาระงานที่เกิดจากผู้ใช้บริการแต่ละเครื่องด้วย ดังนั้นวิธีการที่จะทำให้ระบบสามารถรองรับผู้ให้บริการทุกคนได้อย่างมีประสิทธิภาพมากที่สุดคือการปรับภาระงานให้กับเซิร์ฟเวอร์เก็บข้อมูลทุกๆเครื่องอย่างเหมาะสมที่สุด แต่ในงานวิจัยนี้จะเน้นการทดสอบในระบบที่เซิร์ฟเวอร์แต่ละเครื่องสามารถรับภาระงานได้เท่ากันทั้งระบบก่อน



ภาพที่ 3 ในมุมมองระดับ โครงสร้างพื้นฐานระหว่างผู้ให้บริการและเครื่องแม่ข่ายเก็บข้อมูล

จากแบบจำลองระดับโครงสร้างพื้นฐานที่ได้กล่าวมา ในความเป็นจริงแล้วการเข้าใช้งานระบบเก็บข้อมูลแบบกลุ่มเมฆเกิดจากการเข้าใช้งานของผู้ใช้บริการ ที่ต้องการเข้าถึงวัตถุเก็บข้อมูล (O) โดยวัตถุเก็บข้อมูลแต่ละตัวนั้นจะอยู่ในเซิร์ฟเวอร์เก็บข้อมูล เราสามารถอธิบายความสัมพันธ์ภายในระหว่างผู้ใช้บริการ วัตถุเก็บข้อมูล และเซิร์ฟเวอร์หรือโหนดเก็บข้อมูลได้ดังภาพที่ 4



ภาพที่ 4 ความสัมพันธ์ของวัตถุเก็บข้อมูล เซิร์ฟเวอร์เก็บข้อมูล และผู้ใช้บริการ

จากภาพที่ 4 เมื่อผู้ใช้บริการ (C) มีการเข้าใช้งานวัตถุเก็บข้อมูล (O) ใดๆจะทำให้เกิดภาระงาน (w) ต่อการเข้าใช้งานหนึ่งๆ สำหรับผู้ใช้บริการในขณะหนึ่งๆอาจจะมีการเข้าใช้งานวัตถุเก็บข้อมูลมากกว่าหนึ่งการเข้าใช้งานก็ได้ ซึ่งจะทำให้เกิดภาระงานต่อวัตถุเก็บข้อมูลมากกว่าหนึ่งภาระงานตามมาด้วย หากพิจารณาที่เซิร์ฟเวอร์เก็บข้อมูล (S) จะพบว่าแต่ละเซิร์ฟเวอร์เก็บข้อมูลจะมีวัตถุเก็บข้อมูลที่ได้รับผิดชอบอยู่ภายในที่แตกต่างกัน การที่วัตถุเก็บข้อมูลใดๆได้รับการภาระงานก็เสมือนว่าเซิร์ฟเวอร์เก็บข้อมูลได้รับภาระงานจำนวนนั้นด้วย

ให้วัตถุเก็บข้อมูลแต่ละตัวสามารถสร้างภาระงานให้กับเซิร์ฟเวอร์เก็บข้อมูลในค่าต่างๆกัน ดังนั้นจะได้ว่าภาระงานเซิร์ฟเวอร์เก็บข้อมูลใดๆ มีค่าเท่ากับผลรวมของภาระงานที่สร้างขึ้นโดยวัตถุเก็บข้อมูลทุกๆตัวที่อยู่ในเซิร์ฟเวอร์เก็บข้อมูลนั้นๆ ดังนั้นค่าเฉลี่ยของภาระงานที่เซิร์ฟเวอร์เก็บข้อมูลทุกตัวในระบบต้องรับจะเป็นไปตาม สมการที่ (1) โดยให้ L_i คือภาระงานของเซิร์ฟเวอร์เก็บข้อมูลตัวที่ i โดยในระบบจะมีเซิร์ฟเวอร์เก็บข้อมูลทั้งหมด m เครื่อง

$$L_{avg} = \frac{\sum_{i=1}^m L_i}{m} \quad (1)$$

โดยในการประเมินว่า ณ เวลาหนึ่งๆ ระบบสามารถกระจายภาระงานได้ดีเพียงใด จะพิจารณาจากผลรวมของค่าผลต่างระหว่างทุกๆ L_i กับ L_{avg} ซึ่งถ้าหากผลรวมดังกล่าวยังมีค่าน้อยแสดงว่าระบบสามารถกระจายภาระงานได้ดี ในทางตรงข้ามค่าผลรวมที่มากแสดงถึงการกระจายภาระงานที่ไม่เหมาะสม ดังแสดงไว้ใน สมการ (4) ค่าผลรวมดังกล่าวนี้จะขึ้นอยู่กับรูปแบบการจัดวางวัตถุเก็บข้อมูลที่สัมพันธ์กับเซิร์ฟเวอร์เก็บข้อมูล โดยเราจะให้ความสำคัญดังกล่าวอยู่ในรูปของเวกเตอร์ M โดยจะกล่าวถึงรายละเอียดในภายหลัง

ดังนั้นวิธีการในการกระจายภาระงานให้กับเซิร์ฟเวอร์เก็บข้อมูลหรือโหนดเก็บข้อมูลอย่างเหมาะสม สามารถทำได้โดยการปรับเปลี่ยนการจัดวางวัตถุเก็บข้อมูลให้อยู่ในโหนดเก็บข้อมูลที่เหมาะสมดังที่จะนำเสนอในหัวข้อต่อไป

2. การใช้ขั้นตอนวิธีเชิงพันธุกรรมในการวางแผนการจัดการวัตถุ

วิธีการในการประยุกต์ใช้ขั้นตอนวิธีเชิงพันธุกรรมกับระบบที่ได้นำเสนอไว้จะอธิบายตามขั้นตอนในการทำขั้นตอนวิธีเชิงพันธุกรรม โดยทั่วไป ดังนี้

1. Initialization: สร้างสายโครโมโซม M หรือเวกเตอร์ M ดังสมการที่ (3) ที่มีความแตกต่างกันขึ้นมาเป็นจำนวนมาก โดยความยาวของสายโครโมโซมจะมีขนาดเท่ากับจำนวนของวัตถุเก็บข้อมูลในระบบรวม (n)

2. Selection: เลือกสายโครโมโซมที่สร้างขึ้นมา เพื่อเป็นรุ่นพ่อแม่ในการสร้างรุ่นถัดไปมาจำนวนหนึ่ง โดยใช้ฟังก์ชันความดีจากสมการที่ (4) มาเป็นตัวคัดเลือก

3. Reproduction: ขั้นตอนนี้จะเป็นการสร้างรุ่นลูกขึ้นมาใหม่จากรุ่นพ่อแม่ในรุ่นปัจจุบัน โดยในกระบวนการนี้จะกำหนดให้มีการกลายพันธุ์ของรุ่นลูกเล็กน้อย และในการทำการแลกเปลี่ยนยีน (recombination) จะมีรูปแบบอยู่ 4 รูปแบบดังภาพที่ 5

4. Termination: ในงานวิจัยนี้ได้กำหนดการเสร็จสิ้นของขั้นตอนวิธีเชิงพันธุกรรม โดยกำหนดให้หยุดกระบวนการที่รุ่นที่ 10,000 เพื่อที่จะเปรียบเทียบค่าอื่นๆ ซึ่งจะกล่าวต่อไป

จากภาพที่ 2 ในส่วนการทำงานย่อย object placement manager จะเป็นส่วนที่คอยวางแผนจัดการปรับเปลี่ยนย้ายตำแหน่งของวัตถุเก็บข้อมูล ซึ่งในส่วนการทำงานนี้เราจะสามารถปรับเปลี่ยนโดยใช้ขั้นตอนวิธีเชิงพันธุกรรมเข้ามาเป็นตัวตัดสินใจและพิจารณาวิธีการปรับความสัมพันธ์ระหว่างวัตถุเก็บข้อมูล และ โหนดเก็บข้อมูล ดังนั้นผลจากการทำงานดังกล่าวก็จะทำให้เราได้ storage object movement list ด้วยการใช้ขั้นตอนวิธีเชิงพันธุกรรมนั่นเอง

สำหรับการเชื่อมโยงความสัมพันธ์ระหว่างวัตถุเก็บข้อมูลและ โหนดเก็บข้อมูล จะสามารถเก็บอยู่ในรูปเมตริกซ์ K ดังสมการที่ 2 สมมติให้มี โหนดเก็บข้อมูล m โหนด และมีวัตถุเก็บข้อมูล n ตัว โดยค่า S_{ij} ที่มีค่าเป็น 1 หมายถึงมีวัตถุเก็บข้อมูล j อยู่ในโหนดเก็บข้อมูล i ในทางตรงกันข้ามถ้าหากค่าที่ S_{ij} ใดๆมีค่าเป็น 0 แสดงว่าวัตถุเก็บข้อมูลและ โหนดเก็บข้อมูลนั้นๆไม่มีความสัมพันธ์กัน

$$K = \begin{matrix} & O_1 & O_2 & \dots & O_n \\ \begin{matrix} S_1 \\ S_2 \\ \vdots \\ S_m \end{matrix} & \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix} \quad (2)$$

จากสมการ (1) คือค่าเฉลี่ยของภาระงานในระบบรวม การจะทำให้ระบบกระจายงานอย่างมีประสิทธิภาพจะต้องทำให้โหนดเก็บข้อมูลทุกๆตัว มีภาระงานใกล้เคียงค่าเฉลี่ยที่สุด

$$M = [S_3 \quad S_3 \quad S_1 \quad S_2] \quad (3)$$

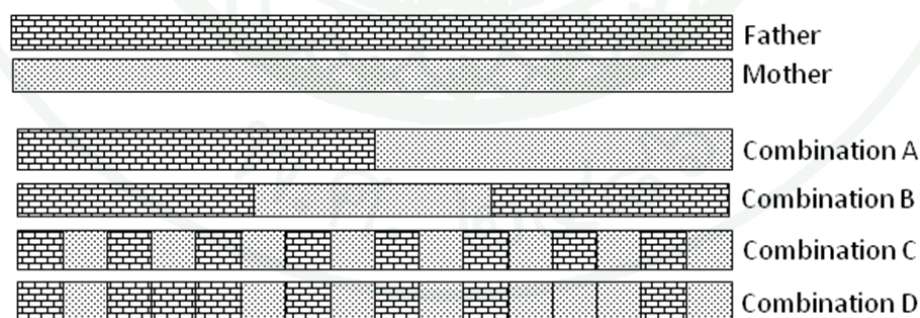
จากสมการที่ (3) แสดงเวกเตอร์ M ซึ่งเป็นเวกเตอร์ที่เก็บลำดับของ S ใดๆที่มีค่า n ตำแหน่ง O_i จากสมการที่ (2) ที่ค่าเป็น 1 โดยเรียงลำดับจาก O_1 ถึง O_n โดยขนาดของเวกเตอร์ M จะมีขนาดเท่ากับ n ในสมการที่ (3) ค่าภายในเวกเตอร์ M เป็นค่าที่จริงอ้างอิงจากค่าของเมตริกซ์ K ในสมการที่ (2)

จากคุณสมบัติของเวกเตอร์ M จากสมการที่ (3) เป็นลำดับการจัดวางของโหนดเก็บข้อมูล โดยจะมีจำนวนสมาชิก S เท่ากับ m ตัว ดังนั้นการจัดวางสับเปลี่ยนตำแหน่งของสมาชิกในเวกเตอร์ M จะเปรียบเสมือนกับการปรับเปลี่ยนความสัมพันธ์ระหว่างวัตถุเก็บข้อมูล และ โหนดเก็บข้อมูล

จากที่กล่าวมาข้างต้นเพื่อประยุกต์การแก้ปัญหาการปรับสมดุลภาระงานด้วยขั้นตอนวิธีเชิงพันธุกรรม จะให้ลำดับของ S ในเวกเตอร์ M เป็นสายโครโมโซมเพื่อใช้ในกระบวนการขั้นตอนวิธีเชิงพันธุกรรม โดยสำหรับการจัดเรียงของ S ในเวกเตอร์ M ในลำดับที่แตกต่างกันเปรียบเสมือนสายโครโมโซมที่มีลักษณะพันธุกรรมที่แตกต่างกัน โดยให้ค่าฟังก์ชันความเหมาะสม (fitness function) เป็นดังสมการที่ (4)

$$D(M) = \sum_{i=0}^n |L_{avg} - L_i| \quad (4)$$

จากสมการที่ (4) จะเป็นฟังก์ชันความเหมาะสม ซึ่งจะเป็นตัวชี้วัดว่าลำดับการวางตัวของเวกเตอร์ M ทำให้การกระจายภาระงานเป็นอย่างไร โดยจากสมการที่ (4) อธิบายได้ว่าเป็นฟังก์ชันที่ให้ค่าผลรวมของผลต่างระหว่างภาระงานของแต่ละโหนดเก็บข้อมูล กับค่าเฉลี่ยภาระงานของระบบรวม ซึ่งถ้าหากค่าผลรวมดังกล่าวมีค่าน้อยก็แสดงว่าโหนดเก็บข้อมูลส่วนใหญ่มีภาระงานใกล้เคียงกับค่าเฉลี่ยของระบบรวม ซึ่งก็หมายถึงความสัมพันธ์ระหว่างวัตถุเก็บข้อมูล และ โหนดเก็บข้อมูลมีความเหมาะสมและทำให้การกระจายภาระงานเป็นไปอย่างมีประสิทธิภาพ



ภาพที่ 5 รูปแบบการแลกเปลี่ยนยีนทั้ง 4 รูปแบบ

จากภาพที่ 5 การรวมกันของโครโมโซมพ่อกับโครโมโซมแม่แบบ A เกิดจากการการแบ่งครึ่งสายโครโมโซมของพ่อครึ่งแรกมาต่อกับครึ่งหลังของโครโมโซมแม่ การรวมแบบ B เกิดจากการแบ่งสายโครโมโซมเป็นสามส่วนแล้วส่วนแรกและส่วนสุดท้ายเป็นของพ่อส่วนตรงกลางเป็นของโครโมโซมแม่ การรวมกับแบบ C เกิดจากการตัดส่วนย่อยของโครโมโซมจากพ่อและแม่มาต่อสลับกัน สำหรับในงานวิจัยนี้คือการตัด S_i ของพ่อและแม่มาต่อสลับกัน และการรวมกันแบบสุดท้ายคือแบบ D จะเป็นการรวมแบบสุ่ม โดย ณ ที่ตำแหน่งใดๆบนโครโมโซมใหม่จะใช้การตัดสินใจแบบสุ่มว่าจะเลือกชิ้นส่วนมาจากโครโมโซมพ่อหรือโครโมโซมแม่

จากวิธีการประยุกต์ใช้วิธีขั้นตอนเชิงพันธุกรรมเพื่อปรับสมดุลภาระงานสำหรับระบบจัดเก็บข้อมูลแบบกลุ่มเมฆที่ได้กล่าวมานั้นจะกล่าวถึงผลการทดลองในหัวข้อผลและวิจารณ์ต่อไป

3. การออกแบบอัลกอริทึมสำหรับการวางแผนจัดวางวัตถุและทำสำเนาแบบพลวัต

จากที่ได้กล่าวในหัวข้อก่อนหน้า เป็นการจัดการโดยการนำขั้นตอนวิธีเชิงพันธุกรรมมาประยุกต์ใช้ ซึ่งการทำงานจะเป็นการย้ายวัตถุเก็บข้อมูล ให้เกิดการกระจายภาระงานเมื่อถึงเวลาที่เหมาะสม แต่เนื่องจากในการทำงานจริงของระบบเก็บข้อมูลแบบกลุ่มเมฆอาจจะไม่จำเป็นต้องมีปรับย้ายวัตถุเก็บข้อมูลทั้งหมด แต่อาจจะสนใจเฉพาะวัตถุเก็บข้อมูลที่มีภาระงานจากการเข้าใช้งานของผู้ใช้สูงก็ได้ เมื่อวัตถุเก็บข้อมูลใดๆมีภาระงานสูงก็อาจจะย้ายวัตถุเก็บข้อมูลนั้นไปในโหนดเก็บข้อมูลที่เหมาะสม

นอกจากนั้นการปรับย้ายวัตถุเก็บข้อมูลเพียงอย่างเดียวก็อาจจะสามารถกระจายภาระงานของโหนดเก็บข้อมูลให้มีความเหมาะสมได้ระดับหนึ่ง การทำสำเนา (replication) ก็เป็นอีกทางหนึ่งในการแก้ปัญหาการกระจายภาระงานอีกทางหนึ่ง ซึ่งก็ต้องแลกมาด้วยการใช้พื้นที่ในการเก็บวัตถุเก็บข้อมูลเพิ่มขึ้น แต่อย่างไรก็ตามหากการเลือกวัตถุเก็บข้อมูลที่จะทำสำเนาที่เหมาะสมก็จะมีประโยชน์ต่อการกระจายภาระงานของระบบรวมอย่างมากเช่นกัน

จากข้อมูลข้างต้นทำให้ทางผู้วิจัยสนใจที่จะนำหลักการดังกล่าวมาประยุกต์ใช้กับการวางแผนจัดวางวัตถุแบบพลวัต โดยจะเลือกเฉพาะวัตถุเก็บข้อมูลที่มีภาระงานหนัก และทำการปรับเปลี่ยนโหนดเก็บข้อมูลเพื่อการกระจายภาระงานที่เหมาะสมยิ่งขึ้น

และเพื่อการกระจายภาระงานที่ดีขึ้นในงานวิจัยนี้ จึงนำเสนอการทำสำเนาข้อมูลควบคู่ไปกับการวางแผนการจัดวางวัตถุ โดยจะยึดหลักการเดิมที่ได้กล่าวมาก่อนหน้านี้ คือวัตถุเก็บข้อมูลที่มีภาระงานสูงจะถือว่าเป็นวัตถุเก็บข้อมูลที่ต้องย้ายตำแหน่ง หรือต้องการการทำสำเนาเพื่อกระจายภาระงาน และจากหลักการดังกล่าวจะทำให้เกิดสำเนาของวัตถุเก็บข้อมูลเพิ่มขึ้นเรื่อยๆในระบบ ในงานวิจัยนี้จึงนำเสนอวิธีการที่จะลดจำนวนสำเนาของวัตถุเก็บข้อมูลที่มีภาระงานรวมน้อยลง โดยใช้วิธีการทำลายสำเนาของวัตถุเก็บข้อมูล โดยจะมีเวลาที่กำหนดไว้ เมื่อถึงเวลาใดๆที่กำหนดไว้สำหรับสำเนาของวัตถุเก็บข้อมูลที่มีอายุครบตามกำหนดของตัวเองก็จะต้องทำลายทิ้ง และหากสำเนาวัตถุเก็บข้อมูลนั้นๆยังมีภาระงานที่สูงอยู่ระบบก็จะทำการสร้างสำเนาของวัตถุเก็บข้อมูลใหม่มาทดแทนเอง

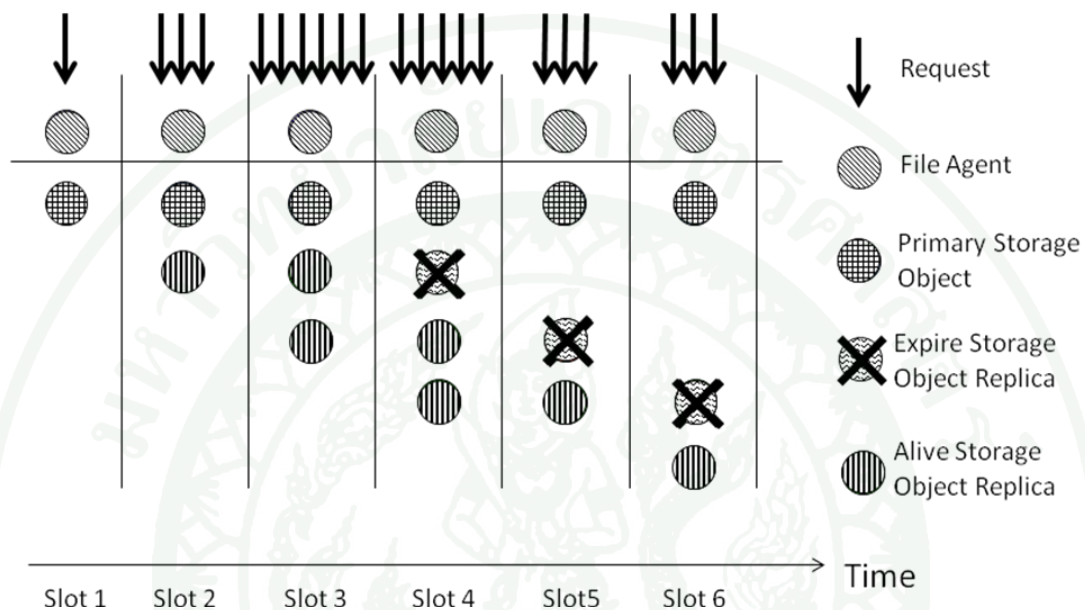
โดยในการจัดวางตำแหน่งสำหรับสำเนาวัตถุข้อมูลที่ได้จากการสำเนาวัตถุเก็บข้อมูลนั้นในการทดลองนี้จะเลือกใช้วิธีการแบบสุ่ม โดยในการสุ่มจะเลือกเฉพาะโหนดเก็บข้อมูลที่ยังไม่มีสำเนาวัตถุเก็บข้อมูลตัวนั้นๆอยู่แล้วเท่านั้น สำหรับในขั้นตอนการเลือกตำแหน่งสำหรับจัดวางสำเนาวัตถุข้อมูลนั้นในงานวิจัยนี้เลือกใช้การจัดวางข้อมูลแบบสุ่มและไม่จัดวางซ้ำเนื่องจากวิธีการนี้จะทำให้การจัดวางวัตถุเก็บข้อมูลหมุนเวียนอย่างทั่วถึงทุกโหนดเก็บข้อมูลโดยอัตโนมัติอยู่แล้วในกรณีที่ต้องการทดลองใช้การจัดวางโดยใช้ขั้นตอนวิธีเชิงพันธุกรรมมาประยุกต์ใช้ก็สามารถทำได้เช่นกัน แต่ก็น่าจะทำให้ผลที่ไม่แตกต่างกันเพราะวัตถุเก็บข้อมูลที่ได้รับการจัดวางในระบบไม่ได้เป็นวัตถุที่ถูกจัดวางอย่างถาวร แต่จะถูกทำลายเมื่อถึงระยะเวลาที่กำหนดไว้ดังได้กล่าวมาแล้ว

โดยในการออกแบบจำลองของการทำสำเนาแบบพลวัตตามที่ได้เสนอมานั้น ทางผู้วิจัยจะกำหนดผลจากการทดสอบโดยวัดจากการกระจายตัวของภาระงานในระบบ ซึ่งก็คือค่าเบี่ยงเบนของการรับภาระงานในโหนดเก็บข้อมูลแต่ละตัว โดยหากระบบรวมมีค่าเบี่ยงเบนน้อยก็แสดงว่าการกระจายตัวของภาระงานในระบบดี ค่าผลลัพธ์อีกค่าที่น่าสนใจก็คือค่าเฉลี่ยของพื้นที่เก็บข้อมูลที่ถูกใช้งาน หากมีการใช้พื้นที่เฉลี่ยของระบบน้อยก็จะเป็นผลดี

ทั้งนี้ค่าตัวแปรที่สำคัญที่ในงานวิจัยนี้สนใจสำหรับการทำสำเนาข้อมูลแบบพลวัต มีอยู่ 2 ค่าก็คือ

1. อายุของสำเนาข้อมูล (L): เป็นระยะเวลาในการคงอยู่ของสำเนาที่ถูกสร้างขึ้นมา โดยเมื่อถึงระยะเวลาที่กำหนดวัตถุเก็บข้อมูลนั้นๆจะต้องถูกทำลายไป

2. คาบเวลาในการทำสำเนา (P): เป็นระยะเวลาที่ใช้ในการตรวจสอบว่าวัตถุเก็บข้อมูลใดควรจะต้องมีการสร้างสำเนาเพิ่มบ้าง และถ้าวัตถุเก็บข้อมูลใดจำเป็นต้องสร้างสำเนาเพิ่มก็จะสร้างทันที โดยการสร้างสำเนาจะนำไปใส่ไว้ในโหนดเก็บข้อมูลที่ยังไม่มีสำเนาของวัตถุเก็บข้อมูลนั้นๆ เท่านั้น



ภาพที่ 6 จำลองสถานการณ์การทำงานของการสำเนาแบบพลวัต

จากภาพที่ 6 ช่วงเวลาที่ 1 จะมีการใช้งานอยู่ 1 หน่วยการใช้งาน โดยจะสมมุติว่าวัตถุเก็บข้อมูล 1 ตัวเพียงพอที่จะรับการใช้งาน 1 หน่วย ต่อมาในช่วงเวลาที่ 2 มีการใช้งานเพิ่มเข้ามาเป็น 3 หน่วย ซึ่งจะสมมุติว่าวัตถุเก็บข้อมูล 1 ตัวไม่สามารถรับภาระงานได้ จึงต้องเพิ่มสำเนาขึ้นมาดังแสดงเป็นวงกลมสีเขียว และถัดมาในช่วงเวลาที่ 3 ก็มีการใช้งานเพิ่มเข้ามาเป็น 6 หน่วย ซึ่งจะสมมุติว่าวัตถุเก็บข้อมูล 2 ตัว ไม่สามารถรับภาระงานได้เพียงพอจึงเกิดการเพิ่มสำเนาขึ้นอีก จนมีวัตถุเก็บข้อมูลทั้งหมด 3 ตัว จากนั้นในช่วงเวลาที่ 4 จำนวนการใช้งานก็ยังมีจำนวนอยู่ 6 หน่วยเท่าเดิม แต่ด้วยข้อกำหนดของระบบที่เราสมมุติให้สำเนาของวัตถุเก็บข้อมูลมีอายุแค่ 2 หน่วยเวลาดังนั้นจากรูปจะเห็นว่ามียวงกลมกากบาทอยู่ ซึ่งหมายความว่าสำเนานั้นๆหมดอายุแล้ว แต่ถ้าหากยังมีการใช้งานจากการใช้งานเก่าอยู่ก็จะรอจนเสร็จ และจะ โคนทำลายจริงเมื่อไม่มีการเข้าใช้งานสำเนานั้นแล้ว ที่ช่วงเวลา 5 จะเห็นว่าสำเนาทั้งหมดอายุในช่วงเวลาที่ 4 โคนทำลายไปแล้ว และก็มีสำเนาที่หมดอายุเพิ่มมาอีกเช่นกัน และเนื่องจากที่ช่วงเวลา 5 นี้มีการใช้งานเพียงแค่ 3 หน่วย จึงมีวัตถุเก็บข้อมูลเพียงแค่ 2 ตัวก็เพียงพอ จึงไม่มีการเพิ่มของวัตถุเก็บข้อมูล ต่อมาในช่วงเวลาที่ 6

วัตถุเก็บข้อมูลตัวเก่าก็หมดอายุ แต่เนื่องจากยังมีการใช้งานเป็นจำนวน 3 หน่วยเช่นเดิม ทำให้มีการสร้างสำเนาเพิ่มอีกหนึ่งตัวไว้รองรับการใช้งานดังกล่าวดังกล่าว

สำหรับขั้นตอนการสำเนานั้นถ้าให้ w_{avg} คือค่าภาระงานเฉลี่ยสำหรับวัตถุเก็บข้อมูลใดๆ และให้ w_{gain} เป็นค่าภาระงานที่เรากำหนดขึ้นมา โดยถ้าหากค่า w_{gain} ยิ่งมีค่ามากก็จะทำให้การตัดสินใจในทำสำเนาจะเกิดขึ้นก็ต่อเมื่อมีการะงานมากขึ้นด้วย โดยจะให้ P_0 เป็นคาบเวลาในการทำงานเพื่อพิจารณาว่าควรจะทำสำเนาหรือไม่สำหรับวัตถุเก็บข้อมูล O ใดๆ เราสามารถอธิบายด้วยรหัสจำลองได้ดังนี้

```

Input:  $w_{gain}, w_{avg}, O$ 
1. for each time period  $P_0$  do
2.   if  $w_{avg} > w_{gain}$ 
3.     Find empty  $OSD_{new}$  for  $O$ 
4.     if can find  $OSD_{new}$ 
5.       Create replica at  $OSD_{new}$ 
6.       Update metadata
7.     end if
8.   end if
9. end for

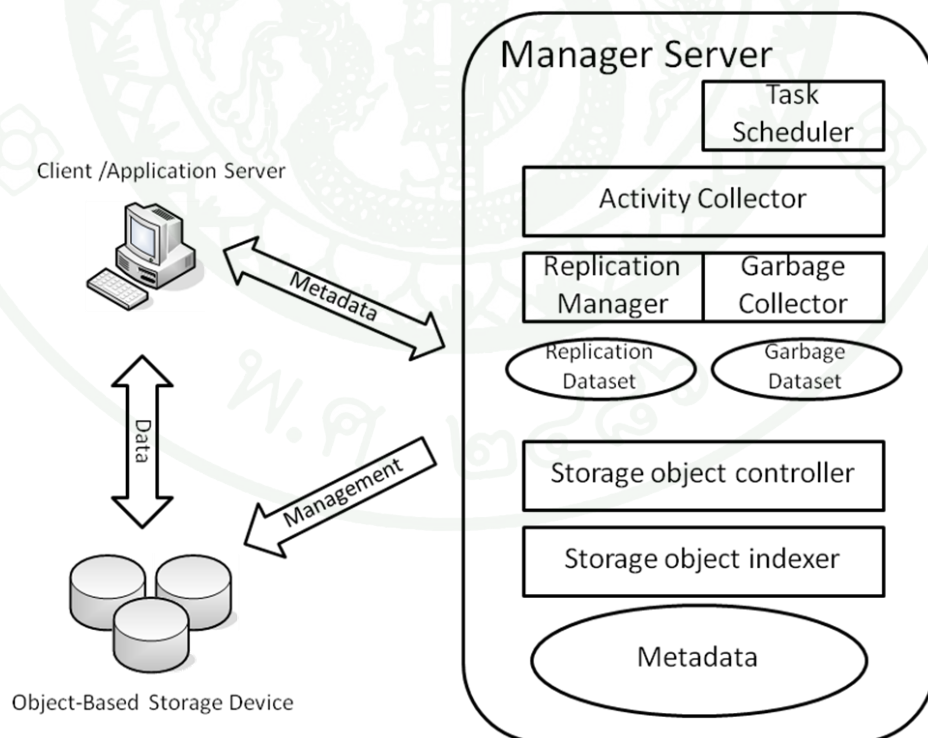
```

สำหรับขั้นตอนการทำลายสำเนาที่หมดอายุทิ้ง ถ้ากำหนดให้ o_i เป็นสำเนาตัวที่ i ของวัตถุเก็บข้อมูลหนึ่งๆ โดยระบบจะทำการพิจารณาการทำลายดังกล่าวทุกๆคาบเวลา L_0 ซึ่งในแต่ละวัตถุเก็บข้อมูลก็สามารถมีค่า L_0 ที่แตกต่างกันได้ตามความเหมาะสม เราสามารถอธิบายด้วยรหัสจำลองได้ดังนี้

Input: $o_i, i; i > 0$

1. **for** each time period L_0 **do**
2. **if** o_i is expired **or** o_i is marked flag
2. **if** o_i has holding request now
3. Mark o_i as need to destroy
4. **else**
5. Destroy o_i
6. **end if**
7. **end if**
8. **end for**

โดยในการปรับเปลี่ยนแบบจำลองเพื่อให้รองรับคุณสมบัติและขั้นตอนการทำงานดังกล่าวมานั้น ทางผู้วิจัยได้มีการพัฒนาแบบจำลองจากแบบเดิม โดยมีรายละเอียดดังภาพที่ 7



ภาพที่ 7 การทำงานย่อยภายในเซิร์ฟเวอร์ควบคุมสำหรับการสำเนาแบบพลวัต

จากภาพที่ 7 ได้นำแบบจำลองเดิมจากภาพที่ 2 มาพัฒนา เพื่อให้สามารถรองรับการทำงานแบบการจัดวางวัตถุและทำสำเนาแบบพลวัตได้ โดยรายละเอียดการทำงานของส่วนการทำงานย่อยภายในแบ่งเป็น 6 ส่วนดังนี้

1. Activity collector ส่วนการทำงานนี้จะทำหน้าที่ในการเก็บข้อมูลรายละเอียดการเข้าถึงวัตถุเก็บข้อมูล โดยข้อมูลที่จะเก็บนั้นจะมีข้อมูลสองส่วนคือ ชื่อระบุตัวตนของวัตถุเก็บข้อมูล และความเร็วการใช้งานโดยประมาณที่ใช้ในการเข้าถึงวัตถุเก็บข้อมูล

2. Replication manager เป็นส่วนที่นำข้อมูลจาก activity collector และ metadata มาประมวลผล ซึ่งข้อมูลดังกล่าวจะทำให้ทราบถึงประมาณภาระงานที่วัตถุเก็บข้อมูลแต่ละตัวรับอยู่ โดยข้อมูลนี้จะเป็นปัจจัยสำคัญที่ช่วยให้ replica manager สร้าง replica dataset ซึ่งเป็นข้อมูลที่จะใช้ในการทำสำเนา

3. Garbage collector ส่วนนี้เป็นส่วนสำคัญสำหรับการทำสำเนาแบบพลวัตของงานวิจัยนี้ เนื่องจาก garbage collector จะใช้ข้อมูลการเข้าถึงข้อมูล และ metadata ในการพิจารณาว่าสำเนาตัวใดควรทำลายออกจากระบบ โดยสำเนาแต่ละตัวจะมีอายุที่โค่นกำหนดไว้ (storage object life-time) ถ้าหากตรวจพบว่าสำเนาใดมีอายุเกินค่าดังกล่าวก็จะตรวจสอบว่ามีการเข้าใช้งานสำเนานั้นอยู่หรือไม่ หากไม่มีก็จะทำลายสำเนานั้นทิ้ง แต่หากมีการเข้าใช้งานอยู่ก็จะบันทึกไว้ว่าสำเนานั้นหมดอายุแล้ว ซึ่งหากมีการร้องขอเข้าใช้งานหลังจากนี้เป็นต้นไป ก็จะไม่อนุญาตให้เข้าถึงสำเนา และในรอบถัดไปที่ garbage collector ทำงาน และตรวจพบสำเนาที่หมดอายุดังกล่าว และไม่มีการใช้งานอยู่ก็จะจัดการทำลายสำเนาที่ออกจากระบบทันที

4. Storage object controller เป็นส่วนการทำงานที่นำข้อมูลจาก replica dataset และ garbage dataset มาใช้ในการดำเนินการปรับย้าย คัดลอก หรือทำลายวัตถุเก็บข้อมูล และสำเนา โดยจะติดต่อสื่อสารโดยตรงกับ โหนดเก็บข้อมูล (object-based storage device) และหลังจากการทำงานทุกครั้งก็จะต้องมีการแจ้งไปยัง storage object indexer เพื่อให้ปรับปรุง metadata ให้ทันสมัยอยู่เสมอ

5. Storage object indexer ข้อมูล metadata จะถูกปรับปรุงแก้ไขโดยส่วนการทำงานนี้ ซึ่งข้อมูลที่เก็บอยู่ใน metadata ได้แก่ storage object metadata ข้อมูลตำแหน่งของสำเนาแต่ละตัว ข้อมูลการเข้าใช้งานสำเนาแต่ละตัว ณ ขณะนั้นๆ รวมไปถึงอายุที่เหลือของสำเนาแต่ละตัวด้วย ในอีก

หน้าที่หนึ่ง storage object indexer ก็จะเป็นหน่วยการทำงานที่ติดต่อกับผู้ใช้งานที่จะเข้ามาขอใช้งานวัตถุเก็บข้อมูลโดยตรงอีกด้วย

6. Task scheduler จะทำหน้าที่เป็นตัวกระตุ้นให้ replication manager และ garbage collector ทำงานเมื่อถึงเวลาที่กำหนด

และทั้งหมดที่อธิบายมานี้คือตัวอย่างการทำงานของการทำงานสำหรับทำสำเนาข้อมูลแบบพลวัตในงานวิจัยนี้

4. สภาพแวดล้อมในการทดสอบ

ในงานวิจัยนี้ได้ทดสอบแนวคิดโดยการจำลองแบบ (Simulation) โดยพัฒนาโปรแกรมจำลองแบบสำหรับงานนี้ด้วยภาษาจาวา ระบบทดสอบจะมีข้อมูลอินพุตเป็นล็อกไฟล์ที่แสดงการเข้าใช้งานไฟล์หรือวัตถุเก็บข้อมูล ดังที่ได้เคยกล่าวมา จากนั้นระบบทดสอบจะนำข้อมูลที่ได้อาประมวลผลเสมือนว่ามีการเข้าใช้งานวัตถุเก็บข้อมูลนั้นๆตามเวลาจริงที่ได้จากล็อกไฟล์ของเครื่อง mirror1.ku.ac.th โดยข้อมูลสำคัญที่ระบบจะลองจะนำมาใช้ในการจำลองสถานการณ์คือหมายเลขไอพีของเครื่องที่เข้ามาใช้งานไฟล์ ชื่อไฟล์ที่เรียกใช้งานหรือดาวน์โหลด เวลา ณ ขณะเริ่มใช้งานไฟล์ และขนาดของไฟล์ใดๆ โดยแต่ละไอพีของเครื่องที่เข้ามาใช้งานไฟล์จะเป็นตัวแทนของผู้ใช้งานหนึ่งๆ ชื่อไฟล์จะเป็นตัวแทนของวัตถุเก็บข้อมูลหนึ่งๆ และเวลาในการเข้าใช้งานจะเปรียบเสมือนเป็นรูปแบบการเข้าใช้งานวัตถุเก็บข้อมูล ซึ่งแต่ละวัตถุเก็บข้อมูลก็จะมีรูปแบบการเข้าใช้งานไฟล์ที่แตกต่างกันไป

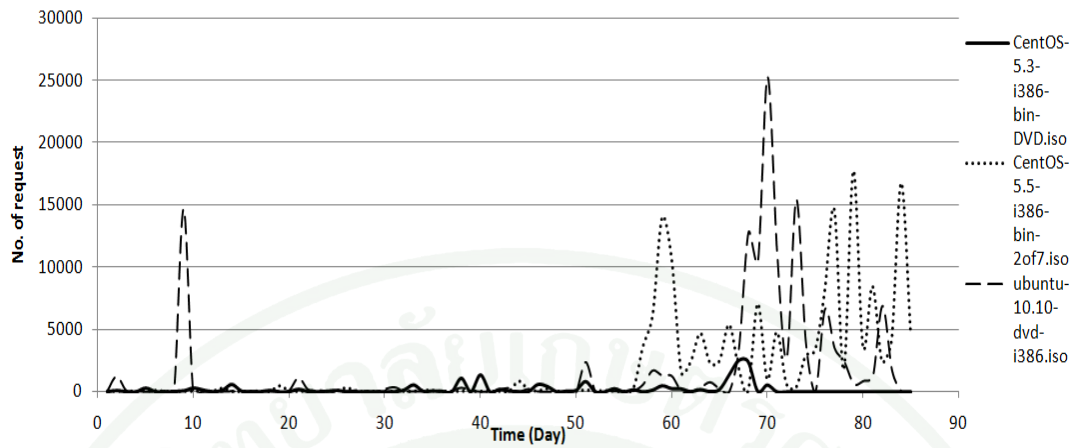
ในงานวิจัยนี้บางการทดลองจะใช้ล็อกไฟล์ที่ได้มาจากการใช้งานจริง และในบางการทดลองก็จะสร้างล็อกไฟล์ขึ้นมาเพื่อใช้ในการทดลองโดยเฉพาะ โดยระบบทดสอบที่ได้สร้างขึ้นมานี้จะจำลองการดาวน์โหลดไฟล์จากผู้ใช้งาน โดยจะนำช่วงเวลาที่เริ่มดาวน์โหลดไฟล์มาจากรหัสล็อกไฟล์มาผ่านกระบวนการจำลองต่างๆ และสามารถดูได้ว่าเสร็จสิ้นการดาวน์โหลดไฟล์เมื่อใด ซึ่งก็จะสามารถวัดผลได้ว่าระบบของเรามีประสิทธิภาพอย่างไร หรือในกรณีของการสำเนาข้อมูลแบบพลวัต ระบบทดสอบก็จะสามารถดูได้ว่ากระบวนการกระจายของสำเนาวัตถุเก็บข้อมูล และภาระงานในช่วงเวลาต่างๆเป็นอย่างไร ทำให้เราสามารถตรวจสอบประสิทธิภาพในการกระจายภาระงานได้

สำหรับการทดลองเรื่องการสำเนาข้อมูลแบบพลวัตนั้นการติดตั้งของระบบจะเริ่มจากระบบที่มีโหนดเก็บข้อมูลหลายโหนดตั้งแต่แรก แต่ล็อกไฟล์ที่ใช้ในการทดสอบนั้นเป็นล็อกไฟล์จากเครื่องแม่ข่ายเพียงเครื่องเดียว ดังนั้นทางผู้วิจัยจึงนำล็อกไฟล์มาแล้วล็อกไฟล์มาทดสอบ โดยทำการกระจายไฟล์ต่างไปยังโหนดเก็บข้อมูลแบบสุ่ม

ในส่วนขั้นตอนในการจัดเตรียมข้อมูลเพื่อจะมาใช้ในการทดลองนั้น มีกระบวนการขั้นตอนดังนี้

1. นำล็อกไฟล์ที่ได้จากเครื่องแม่ข่าย mirror1.ku.ac.th มาวิเคราะห์เพื่อเลือกเอาเฉพาะล็อกของการเข้าถึงไฟล์ที่น่าสนใจเท่านั้น เนื่องจากปริมาณไฟล์ในล็อกไฟล์มีจำนวนมาก ทางผู้วิจัยได้เลือกไฟล์ที่มีการเรียกใช้งานในปริมาณที่สูงในช่วงระยะเวลาที่กำหนดเท่านั้น
2. สร้างล็อกไฟล์ขึ้นมาใหม่ตามช่วงเวลาที่กำหนดจากไฟล์ที่ได้เลือกไว้แล้วจากขั้นตอนก่อนหน้า
3. นำล็อกไฟล์ใหม่ที่ได้เป็นอินพุตสำหรับนำไปทดสอบในแบบจำลองที่ได้สร้างขึ้นมา

จากภาพที่ 8 ทางผู้วิจัยได้นำล็อกไฟล์จากเครื่องแม่ข่าย mirror1.ku.ac.th โดยเลือกไฟล์ที่ได้ได้รับความนิยมในการใช้งานมาแสดงในรูปแบบกราฟเพื่อแสดงให้เห็นว่าแต่ละไฟล์มีความนิยมในการการเข้าใช้งานที่แตกต่างกันตามช่วงเวลา โดยล็อกไฟล์ที่ได้มาเป็นการเก็บข้อมูลตั้งแต่วันที่ 28 พฤศจิกายน 2553 จนถึงวันที่ 20 กุมภาพันธ์ 2554



ภาพที่ 8 ตัวอย่างจำนวนการเข้าใช้งานไฟล์จากสื่อไฟล์จาก mirror1.ku.ac.th

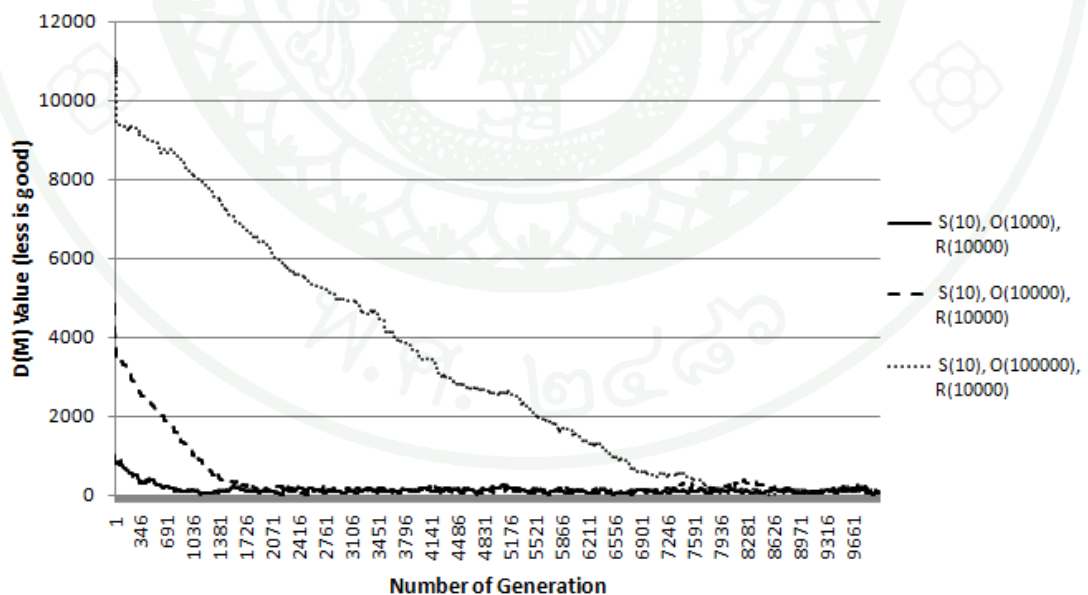
ผลและวิจารณ์

ผล

ในงานวิจัยนี้จะมีการทดลองหลักในส่วนของการทำการวางแผนการจัดวางวัตถุอยู่สองส่วนด้วยกัน คือการนำขั้นตอนวิธีเชิงพันธุกรรมมาประยุกต์ใช้ และการทำสำเนาแบบพลวัต ซึ่งจะอธิบายผลและวิเคราะห์ที่ได้ดังนี้

1. การทดลองใช้ขั้นตอนวิธีเชิงพันธุกรรมในการวางแผนการจัดวางวัตถุ

ในส่วนการทดลองนี้ได้สร้างสไลด์ไฟลด์ที่มีการกระจายตัวของการเข้าใช้งานวัตถุเก็บข้อมูลเป็นแบบเอกรูปและกำหนดขนาดวัตถุเก็บข้อมูลเป็น 5 ขนาด โดยกำหนดจำนวนของโหนดเก็บข้อมูลให้เป็นค่าคงที่โดยกำหนดให้มีจำนวนโหนดเก็บข้อมูลจำนวน 10 โหนดและจะเปรียบเทียบผลของการกระจายภาระงานดังที่ได้กล่าวมาในสมการที่ (4) โดยจะทดลองเปลี่ยนจำนวนของวัตถุเก็บข้อมูล เพื่อเปรียบเทียบว่าเมื่อจำนวนวัตถุเก็บข้อมูลเพิ่มขึ้นจะมีผลอย่างไรบ้าง

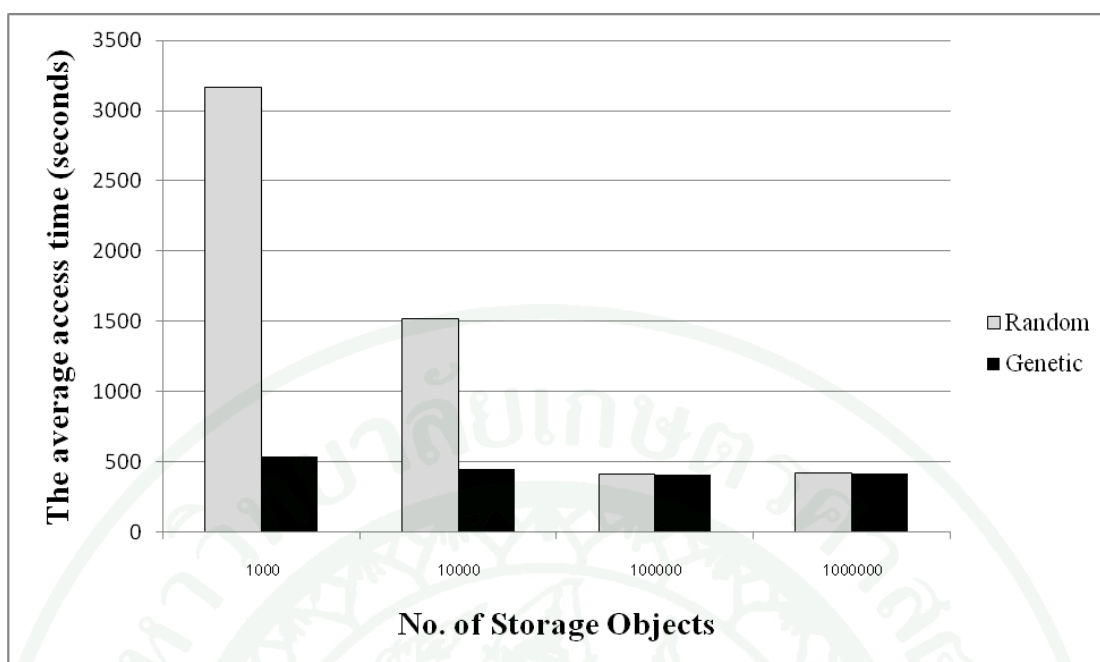


ภาพที่ 9 กราฟเปรียบเทียบค่าของฟังก์ชันค่าความเหมาะสมที่จำนวนวัตถุเก็บข้อมูลต่างๆ กัน

จากภาพที่ 9 เมื่อ S คือจำนวนโหนดเก็บข้อมูล ซึ่งจะมีจำนวนโหนดเก็บข้อมูลคงที่อยู่ที่ 10 โหนด ให้ O เป็นจำนวนวัตถุเก็บข้อมูลและให้ R เป็นจำนวนของรุ่นในการทำขั้นตอนวิธีเชิงพันธุกรรม ซึ่งในการทดลองนี้ก็จะกำหนดไว้เป็นค่าคงที่ที่ 10,000 รุ่น

จากกราฟจะเห็นได้ว่าการเพิ่มจำนวนวัตถุเก็บข้อมูล จุดที่ค่าของฟังก์ชันความเหมาะสมมีค่าเข้าใกล้ 0 จะมีค่าจำนวนรุ่นในการทำขั้นตอนวิธีเชิงพันธุกรรมที่เพิ่มขึ้น ซึ่งก็เป็นพฤติกรรมปกติอยู่แล้ว เนื่องจากการที่มีวัตถุเก็บข้อมูลเป็นปริมาณที่มากขึ้นก็จะทำให้สายโครโมโซมที่ยาวขึ้น การผสมกันระหว่างโครโมโซมพ่อกับแม่แล้วจะได้รุ่นลูกที่ดีก็ทำได้ยากขึ้น ต้องใช้เวลาที่มากขึ้น แต่อย่างไรก็ตามจากการทดลองนี้จะเห็นได้ว่าจำนวนรุ่นของการทำขั้นตอนวิธีเชิงพันธุกรรมนั้น ไม่ได้เพิ่มขึ้นเป็นจำนวนเท่าตามจำนวนของวัตถุเก็บข้อมูลที่เพิ่มขึ้น ถ้าสังเกตที่จำนวนวัตถุเก็บข้อมูล ณ ขณะที่มีจำนวนเท่ากับ 1,000 จะพบว่าค่าของฟังก์ชันความเหมาะสมมีค่าคงที่ในช่วงประมาณรุ่นที่ 800 ถ้าเปรียบเทียบกับในขณะที่มีจำนวนวัตถุเก็บข้อมูลมีจำนวนเท่ากับ 100,000 ค่าของฟังก์ชันความเหมาะสมจะเริ่มมีค่าคงที่อยู่ที่ประมาณรุ่นที่ 8,000 จากการสังเกตดังกล่าวจะเห็นได้ว่าจำนวนรุ่นของการทำขั้นตอนวิธีเชิงพันธุกรรมนั้น ไม่ได้เพิ่มขึ้นตามจำนวนเท่าของจำนวนวัตถุเก็บข้อมูล ซึ่งก็แสดงให้เห็นว่าการนำขั้นตอนวิธีเชิงพันธุกรรมในรูปแบบที่ได้กล่าวมานั้น มาใช้ประยุกต์ใช้กับการทำการวางแผนการจัดวางวัตถุในระบบเก็บข้อมูลแบบกลุ่มเมฆที่ใช้ในงานวิจัยนี้มีผลการทดลองที่ดีพอสมควร

นอกจากนั้นในการทดลองนี้ยังได้มีการเปรียบเทียบให้เห็นถึงประสิทธิภาพในการกระจายภาระงาน โดยใช้ขั้นตอนวิธีการเชิงพันธุกรรม เทียบกับการที่ไม่มีการทำการกระจายภาระงานอีกด้วย

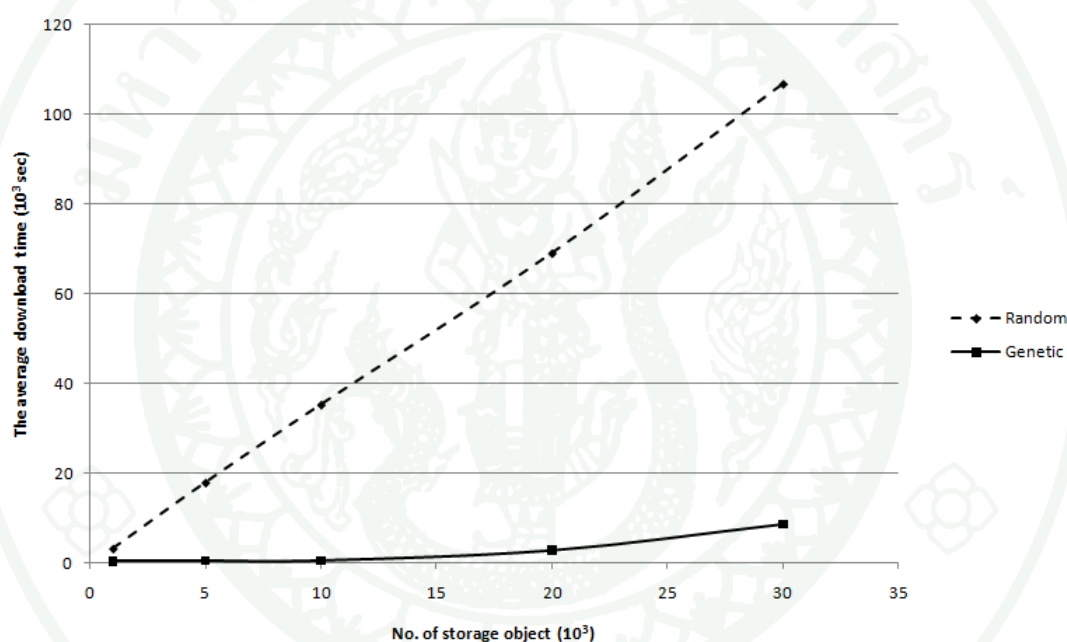


ภาพที่ 10 เวลาเฉลี่ยในการดาวน์โหลดไฟล์เมื่อใช้ขั้นตอนวิธีเชิงพันธุกรรมและแบบสุ่ม

จากภาพที่ 10 เป็นการเปรียบเทียบระยะเวลาในการดาวน์โหลดไฟล์หรือวัตถุเก็บข้อมูล ซึ่งจากกราฟจะแสดงให้เห็นว่าจำนวนวัตถุเก็บข้อมูลที่เพิ่มขึ้นว่าจะมีผลอย่างไรกับระยะเวลาในการดาวน์โหลดไฟล์ โดยได้เปรียบเทียบระหว่างการขั้นตอนวิธีเชิงพันธุกรรมกับการทำการจัดวางวัตถุโดยการสุ่ม โดยในการทดลองนี้จะกำหนดจำนวนโหนดเก็บข้อมูลไว้ที่ 10 โหนดเช่นเดิม และกำหนดให้จำนวนการใช้งานมีค่าคงที่คือมีจำนวน 1,000,000 การใช้งาน ซึ่งจากผลการทดลองที่ได้ในกราฟจะเห็นว่าการใช้ขั้นตอนวิธีเชิงพันธุกรรมในการวางแผนการจัดวางวัตถุ จะช่วยลดเวลาเฉลี่ยในการดาวน์โหลดไฟล์ได้ดีพอสมควร ทั้งนี้จะเห็นว่าที่จำนวนวัตถุเก็บข้อมูลน้อย จะเห็นความแตกต่างที่ชัดเจนกว่าที่จำนวนวัตถุเก็บข้อมูลที่มีจำนวนมากขึ้น ทั้งนี้จริงๆแล้วก็เนื่องจากการทดลองเราได้กำหนดให้จำนวนการเข้าใช้งานมีค่าคงที่คือ 1,000,000 การใช้งาน ซึ่งในการทดลองที่มีวัตถุเก็บข้อมูลจำนวนน้อยกว่าการเข้าใช้งานมาก ก็จะทำให้มีวัตถุเก็บข้อมูลบางตัวที่รับภาระงานหนัก และส่งผลให้โหนดเก็บข้อมูลที่มีวัตถุเก็บข้อมูลที่รับภาระงานหนักอยู่รวมกันหลายตัว ไม่สามารถตอบสนองการดาวน์โหลดได้อย่างมีประสิทธิภาพ เมื่อมีการทำการจัดวางวัตถุโดยใช้ขั้นตอนวิธีเชิงพันธุกรรม ก็จะช่วยให้วัตถุเก็บข้อมูลที่มีภาระงานหนักเหล่านั้นได้กระจายตัวไปตามโหนดเก็บข้อมูลที่เหมาะสม ทำให้เห็นว่าสามารถลดระยะเวลาในการดาวน์โหลดได้อย่างชัดเจน ในทางตรงกันข้ามการทดลองที่มีจำนวนวัตถุเก็บข้อมูลเป็นจำนวนมากซึ่งใกล้เคียงกับจำนวนของการเข้าใช้งาน ก็เปรียบเสมือนว่าการเข้าใช้งานเหล่านั้นได้กระจายไปยังวัตถุเก็บข้อมูล

ในระบบอย่างทั่วถึง จึงเสมือนว่าเป็นการกระจายภาระงานไปโดยปริยาย ทำให้ผลการเปรียบเทียบของการทำการวางแผนการจัดวางวัตถุทั้งสองแบบแทบจะไม่แตกต่างกัน

จากสาเหตุดังกล่าวทางผู้วิจัยจึงได้ทำการทดลองเพิ่มเติมเพื่อพิสูจน์ว่าในกรณีที่จำนวนวัตถุเก็บข้อมูลและการเข้าใช้งานมีปริมาณที่เพิ่มขึ้นอย่างสัมพันธ์จะทำให้การประยุกต์ใช้วิธีขั้นตอนเชิงพันธุกรรมยังสามารถใช้เพื่อช่วยในการปรับสมดุลภาระงานได้ผลเช่นเดิมหรือไม่ โดยในการทดลองจะกำหนดให้การเข้าใช้งานเป็น 10 เท่าของจำนวนวัตถุเก็บข้อมูลเสมอ ซึ่งได้ผลการทดลองดังภาพที่ 11



ภาพที่ 11 เวลาเฉลี่ยในการดาวน์โหลดไฟล์เมื่อมีการเพิ่มจำนวนวัตถุเก็บข้อมูลและการใช้งาน

จากภาพที่ 11 จะเห็นว่าระยะเวลาเฉลี่ยในการดาวน์โหลดไฟล์จะมีค่าน้อยหลังจากการจัดวางวัตถุเก็บข้อมูลด้วยขั้นตอนวิธีเชิงพันธุกรรมแล้ว ทั้งนี้เนื่องจากโดยปกติแล้วหากมีวัตถุเก็บข้อมูลที่มีภาระงานมากอยู่ที่โหนดเก็บข้อมูลใดก็จะมีภาระงานที่ร้องขอข้อมูลที่ค้างต่อกันไปเรื่อยๆ และถ้าหากวัตถุเก็บข้อมูลที่มีภาระงานมากเหล่านั้นอยู่รวมกัน ในโหนดเก็บข้อมูลเดียวกันเป็นจำนวนมากแล้ว ก็จะทำให้เวลาในการดาวน์โหลดข้อมูลเพิ่มมากขึ้น จากกราฟที่ได้แสดงในภาพที่ 11 จะเห็นว่ากราฟของเวลาเฉลี่ยในการดาวน์โหลดของการจัดวางวัตถุเก็บข้อมูลแบบสุ่มจะมีลักษณะเพิ่มขึ้นเรื่อยๆ ค่อยข้างแปรผันตรงกับจำนวนการร้องขอและจำนวนวัตถุเก็บข้อมูล แต่สำหรับ

กราฟเวลาเฉลี่ยในการดาวน์โหลดของการจัดวางวัตถุเก็บข้อมูลที่ใช้ขั้นตอนวิธีเชิงพันธุกรรมในการจัดวางวัตถุเก็บข้อมูลนั้น จะเห็นว่าเวลาเฉลี่ยในการดาวน์โหลดมีแนวโน้มการที่เพิ่มขึ้นสูงในช่วงที่จำนวนการร้องขอและจำนวนวัตถุเก็บข้อมูลเพิ่มขึ้น ทั้งนี้เกิดจากในการทดลองไม่ได้มีการเพิ่มจำนวนโหนดเก็บข้อมูลที่เพิ่มขึ้นด้วย จึงทำให้ข้อจำกัดในการให้บริการของโหนดเก็บข้อมูลโดยรวมลดลง

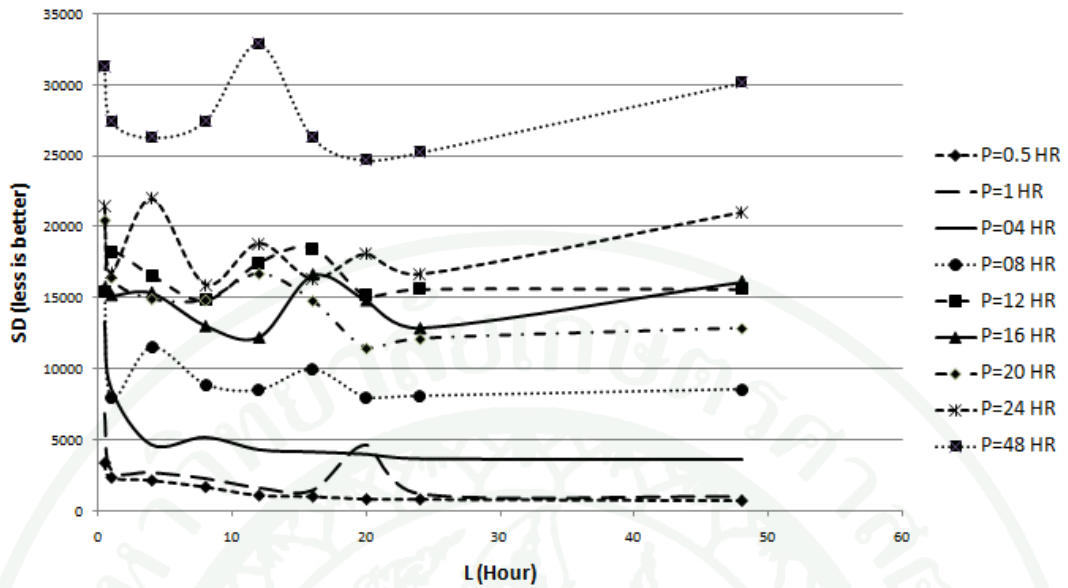
2. การทดลองใช้การจัดวางและการทำงานวัตถุแบบพลวัตร

จากการทดลองถึงแม้ว่าการเปรียบเทียบเวลาเฉลี่ยในการดาวน์โหลดจะเห็นผลชัดว่าการใช้ขั้นตอนวิธีเชิงพันธุกรรมมาประยุกต์ใช้ในการทำการวางแผนการจัดวางวัตถุจะช่วยลดระยะเวลาเฉลี่ยในการดาวน์โหลดไฟล์ แต่ก็ยังมีประเด็นที่น่าสนใจในการทำการวางแผนการจัดวางวัตถุแบบพลวัตร รวมไปถึงการทำงานให้กับวัตถุเก็บข้อมูลที่มีภาระงานหนักด้วย

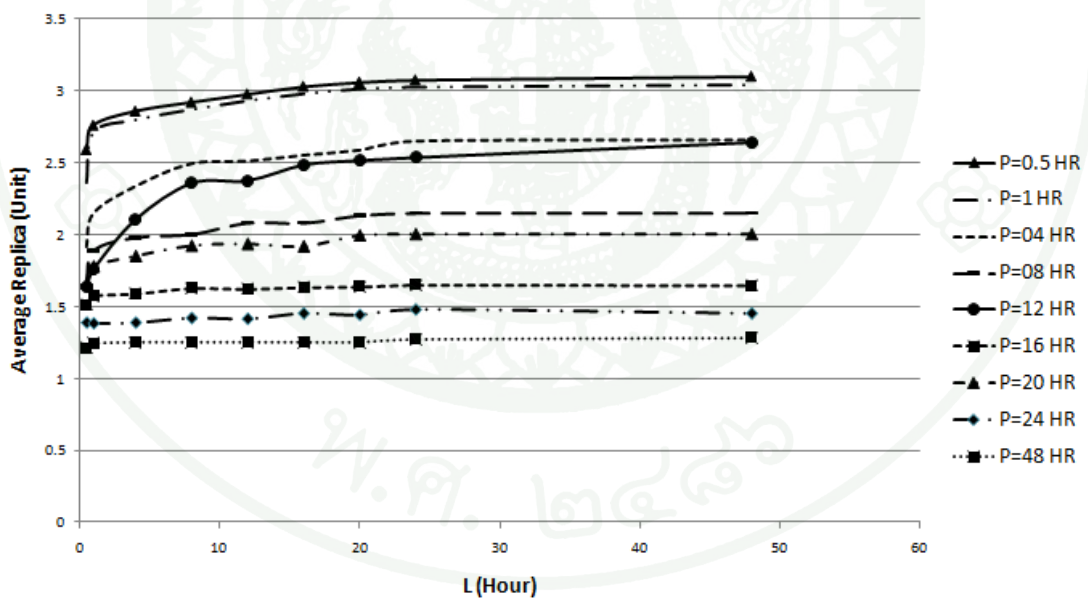
การทำการวางแผนจัดวางวัตถุแบบพลวัตรจะทำให้การใช้งานระบบเก็บข้อมูลแบบกลุ่มเมฆคุ้มค่ามากขึ้น เนื่องจากเกิดจากแนวความคิดที่ว่าวัตถุเก็บข้อมูลใดๆที่มีความนิยมในการใช้งานสูงก็ควรจะถูกย้ายไปอยู่ในโหนดเก็บข้อมูลที่ยังสามารถรองรับภาระงานได้ เพื่อเป็นการเฉลี่ยภาระงานในระบบ แต่อย่างไรก็ตามหากภาระงานที่เกิดขึ้นกับวัตถุเก็บข้อมูลนั้นมากเกินไปที่โหนดเก็บข้อมูลนั้นๆจะรับได้ วิธีเดียวที่จะแก้ปัญหาได้คือการทำสำเนาให้กับวัตถุเก็บข้อมูลนั้นและเลือกโหนดเก็บข้อมูลที่เหมาะสมในการย้ายสำเนานั้นไป

สำหรับผลการทดลองในเรื่องการทำงานแบบพลวัตรนั้นในงานวิจัยนี้ได้ออกแบบมาโดยแบ่งการทดลองออกเป็น 2 การทดลองดังนี้

การทดลองที่ 1 กำหนดค่าคาบเวลาการทำงานสำเนา (P) ให้เป็นค่าคงที่ แล้วทดลองปรับค่าอายุของวัตถุเก็บข้อมูล (L) โดยกราฟแต่ละเส้นจะมีค่า P ต่างๆกัน เพื่อสังเกตผลลัพธ์ของการกระจายตัวของภาระงาน โดยดูจากค่าเบี่ยงเบนของการกระจายตัว และค่าเฉลี่ยการใช้พื้นที่เก็บข้อมูล ซึ่งค่าเบี่ยงเบนของการกระจายภาระงานได้ผลดังภาพที่ 12 และค่าเฉลี่ยพื้นที่ที่ใช้ในการเก็บข้อมูลได้ผลดังภาพที่ 13



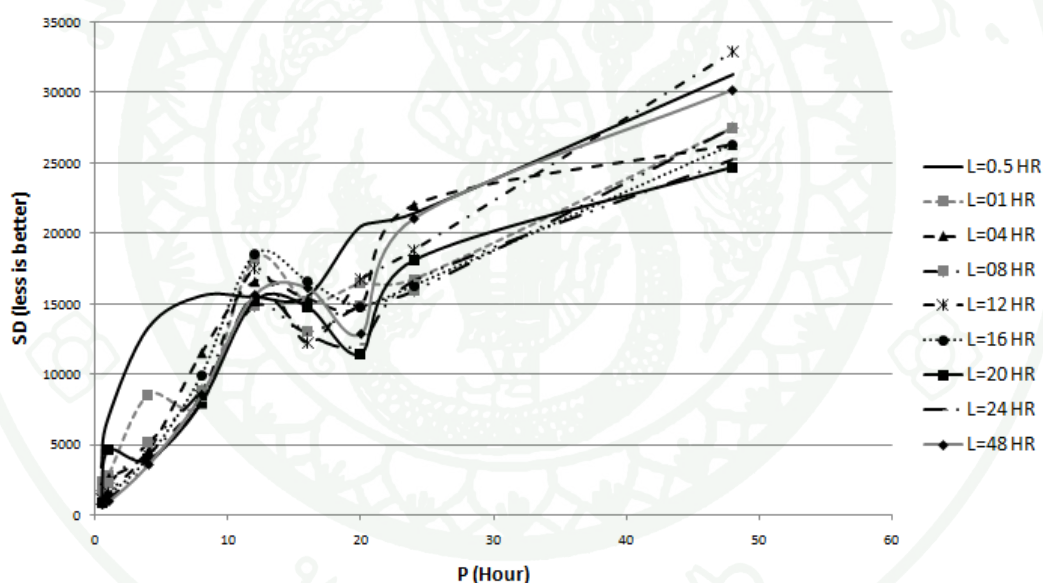
ภาพที่ 12 เปรียบเทียบค่าเบี่ยงเบนการกระจายภาระงานเมื่อปรับค่า L และให้ P คงที่



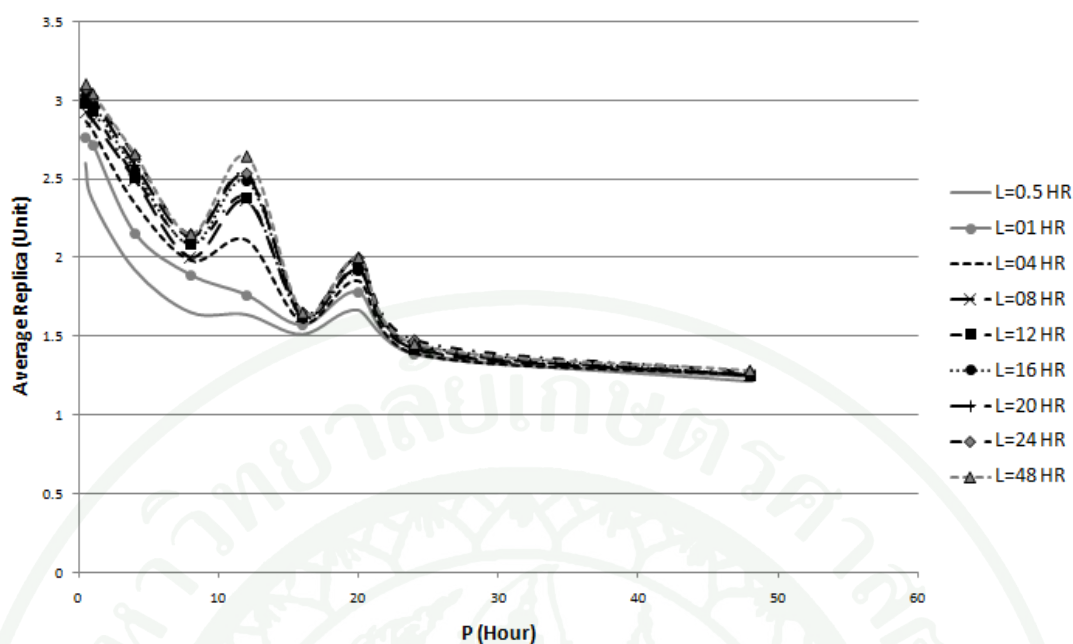
ภาพที่ 13 เปรียบเทียบค่าเฉลี่ยการใช้พื้นที่เก็บข้อมูลเมื่อปรับค่า L และให้ P คงที่

จากภาพที่ 12 จะสังเกตได้ว่าค่า α ค่า P ต่างๆ นั้นค่า L ที่น้อยเกินไปก็ไม่ได้ทำให้การกระจายตัวของภาระงานดี (ค่าเบี่ยงเบนสูง) และ L ที่มากเกินไปก็ไม่ได้ทำให้การกระจายตัวของภาระงานดีเช่นกัน แต่จะมีบางช่วงของ L ที่จะมีค่าเบี่ยงเบนน้อยหรือการกระจายตัวดี แต่สำหรับค่าเฉลี่ยของการใช้งานพื้นที่เก็บข้อมูลจะแปรผันตรงกับค่า L ยิ่งค่า L มากขึ้นพื้นที่เฉลี่ยในการเก็บข้อมูลก็เพิ่มขึ้นเช่นกันจนถึง L ค่าหนึ่ง ซึ่งการเพิ่มค่า L หลังจากจุดดังกล่าว การเพิ่มขึ้นของพื้นที่เฉลี่ยในการเก็บข้อมูลก็ไม่ได้เพิ่มขึ้นมากนัก ดังภาพที่ 13

การทดลองที่ 2 กำหนด L เป็นค่าคงที่ แล้วทดลองปรับค่า P เพื่อสังเกตผลลัพธ์ของการกระจายตัวของภาระงาน โดยดูจากค่าเบี่ยงเบนของการกระจายตัว และค่าเฉลี่ยการใช้พื้นที่เก็บข้อมูล ซึ่งค่าเบี่ยงเบนของการกระจายภาระงานได้ผลดังภาพที่ 14 และค่าเฉลี่ยพื้นที่ที่ใช้ในการเก็บข้อมูลได้ผลดังภาพที่ 15



ภาพที่ 14 เปรียบเทียบค่าเบี่ยงเบนการกระจายภาระงานเมื่อปรับค่า P และให้ L คงที่



ภาพที่ 15 เปรียบเทียบค่าเฉลี่ยการใช้พื้นที่เก็บข้อมูลเมื่อปรับค่า P และให้ L คงที่

จากภาพที่ 14 สังเกตได้ว่าเมื่อ P เพิ่มขึ้นค่าเบี่ยงเบนก็มีแนวโน้มที่เพิ่มขึ้นเช่นกัน ซึ่งก็หมายถึงการกระจายภาระงานที่แผ่ลง ในทางกลับกันค่าเฉลี่ยของพื้นที่เก็บข้อมูลก็จะลดลงเมื่อ P เพิ่มขึ้นดังภาพที่ 15

วิจารณ์

จากผลการทดลองการทำการวางแผนการจัดวางวัตถุเป็นวิธีการที่สามารถเพิ่มประสิทธิภาพของระบบจัดเก็บข้อมูลแบบกลุ่มเมฆได้ในระดับหนึ่ง โดยจากการทดลองในงานวิจัยนี้พบว่าเราสามารถนำขั้นตอนวิธีเชิงพันธุกรรมมาช่วยในการแก้ปัญหาการจัดวางวัตถุในระบบจัดเก็บข้อมูลแบบกลุ่มเมฆให้เหมาะสมได้ และได้ผลลัพธ์ที่มีประสิทธิภาพ แต่อย่างไรก็ตามการทำการวางแผนการจัดวางวัตถุก็มีข้อจำกัดในการกระจายภาระงานอยู่ในกรณีที่มีภาระงานมากระจุกตัวอยู่ที่วัตถุเก็บข้อมูลตัวใดตัวหนึ่งมากเกินไป เนื่องจากอาจจะทำให้ภาระงานนั้นเกินขีดความสามารถของโหนดเก็บข้อมูลได้

ทางผู้วิจัยจึงแก้ปัญหาดังกล่าวด้วยการใช้วิธีการทำสำเนาให้กับวัตถุเก็บข้อมูลที่มีภาระงานสูง เพื่อเป็นการกระจายภาระงานออกไปให้โหนดเก็บข้อมูลอื่นๆ แต่ทั้งนี้การทำสำเนาข้อมูลถึงแม้ว่าจะช่วยให้การกระจายภาระงานมีประสิทธิภาพที่ดีขึ้น แต่ก็จะต้องใช้พื้นที่ในการเก็บข้อมูลเพิ่มจากเดิมเช่นกัน ในงานวิจัยนี้จึงเสนอแนวคิดเรื่องอายุของวัตถุเก็บข้อมูลขึ้นมาเพื่อลดปัญหาการใช้พื้นที่เก็บข้อมูลมากเกินไปจนเกิดความจำเป็น ซึ่งคาบเวลาในการทำสำเนาข้อมูล และค่าของอายุของวัตถุเก็บข้อมูลของระบบก็ขึ้นอยู่กับรูปแบบการใช้งานวัตถุเก็บข้อมูล ซึ่งในงานวิจัยนี้ได้ใช้รูปแบบการเข้าถึงไฟล์หรือวัตถุเก็บข้อมูลจากเครื่องแม่ข่าย mirror1.ku.ac.th

โดยจากผลการทดลองในงานวิจัยนี้พบว่าค่าคาบเวลาในการทำสำเนาข้อมูล (P) จะมีผลต่อการกระจายภาระงานคือยิ่งคาบเวลามากการกระจายภาระงานก็จะยิ่งไม่ดี และยิ่งคาบเวลามากจะส่งผลให้ค่าเฉลี่ยของจำนวนสำเนาข้อมูลน้อยลง ในส่วนของค่าอายุของวัตถุเก็บข้อมูล (L) สำหรับการทดลองจากล็อกไฟล์ในงานวิจัยนี้จะมีผลต่อการกระจายภาระงานและจำนวนสำเนาข้อมูลน้อยกว่าค่าคาบเวลาในการทำสำเนา โดยค่าอายุของวัตถุเก็บข้อมูลทีมากจะทำให้ค่าเฉลี่ยของจำนวนสำเนาข้อมูลยิ่งมากขึ้น แต่สำหรับในเรื่องของการกระจายภาระงานนั้นจะเห็นผลในช่วงที่ค่าอายุวัตถุเก็บข้อมูลมีค่าน้อยๆการเพิ่มค่าอายุของวัตถุเก็บข้อมูลจะส่งผลให้การกระจายภาระงานดีขึ้น แต่อย่างไรก็ตามหลังจากการเพิ่มค่าอายุวัตถุเก็บข้อมูลไปเรื่อยๆยังไม่เห็นแนวโน้มที่ชัดเจนเหมือนกับการปรับค่าคาบเวลาการทำสำเนาข้อมูล

สรุปและข้อเสนอแนะ

สรุป

ในวิทยานิพนธ์นี้ได้นำเสนอวิธีการเพิ่มประสิทธิภาพของระบบจัดเก็บข้อมูลแบบกลุ่มเมฆ โดยนำวิธีในการจัดวางวัตถุเก็บข้อมูลและการทำสำเนาของวัตถุเก็บข้อมูลมาช่วยในการปรับสมดุลภาระงาน โดยแบ่งเป็น 2 วิธีดังนี้

1. การจัดวางวัตถุเก็บข้อมูลโดยใช้ขั้นตอนวิธีเชิงพันธุกรรมมาประยุกต์ใช้ในการจัดวางเพื่อปรับสมดุลภาระงาน ผลลัพธ์ที่ได้วิธีการนี้สามารถช่วยปรับสมดุลการกระจายภาระงานในระบบได้ เหมาะกับระบบที่มีผู้ใช้ปริมาณไม่มากนักเนื่องจากถึงแม้ว่าจะมีการจัดวางวัตถุเก็บข้อมูลที่เหมาะสมแล้ว แต่อย่างไรก็ตามโหนดเก็บข้อมูลก็มีข้อจำกัดในการให้บริการหากมีวัตถุเก็บข้อมูลที่มีการเข้าใช้งานสูงเกินกว่าที่โหนดเก็บข้อมูลจะรับได้ก็จะเกิดปัญหาภาระงานไม่สมดุลได้อีก

2. การทำสำเนาข้อมูลแบบพลวัตเพื่อปรับสมดุลภาระงาน ในงานวิจัยนี้ได้นำเสนอวิธีการในการสำเนาข้อมูลแบบพลวัตโดยอาศัยภาระงานของแต่ละวัตถุเก็บข้อมูลเป็นปัจจัยในการตัดสินใจเพื่อทำสำเนาข้อมูล โดยวัตถุเก็บข้อมูลใดๆจะถูกพิจารณาทำสำเนาข้อมูลเพื่อถึงคาบเวลาที่กำหนด และจัดวางสำเนานั้นไปยังโหนดเก็บข้อมูลที่ยังไม่มีสำเนานั้น วิธีการนี้จะทำให้สามารถกระจายภาระงานไปยังโหนดเก็บข้อมูลอื่นได้ แต่ก็มีผลกระทบตามมาเกี่ยวกับพื้นที่ในการเก็บข้อมูลที่ต้องสูญเสียไปจากสำเนาข้อมูลที่เพิ่มขึ้น ทางผู้วิจัยจึงได้นำเสนอวิธีการที่จะมาควบคุมปริมาณของสำเนาข้อมูลโดยใช้การกำหนดอายุของสำเนาวัตถุเก็บข้อมูลเพื่อทำให้เกิดการทำลายสำเนาวัตถุเก็บข้อมูลนั้นๆทิ้งเมื่อสำเนานั้นๆหมดอายุ จากที่กล่าวมาจะพบว่ามีความคุ้มค่าแปรที่สามารถปรับเปลี่ยนได้สำหรับวัตถุเก็บข้อมูลใดๆอยู่ 3 ค่า คือ เกณฑ์ภาระงานในการตัดสินใจเพื่อทำสำเนาข้อมูล คาบเวลาในการพิจารณาเพื่อทำสำเนาวัตถุข้อมูล และอายุของสำเนาวัตถุเก็บข้อมูลที่สามารถอยู่ในระบบได้ ซึ่งในงานวิจัยจะสนใจค่าคาบเวลาในการตัดสินใจเพื่อทำสำเนาข้อมูล และค่าอายุของสำเนาวัตถุเก็บข้อมูล โดยได้นำเสนอความสัมพันธ์ไปในผลการทดลองแล้ว

ข้อเสนอแนะ

ในการนำแบบจำลองและวิธีการที่ได้นำเสนอในเรื่องการทำสำเนาแบบพลวัตร์ไปใช้งานจริง มีความเป็นไปได้ที่ระบบเก็บข้อมูลแบบกลุ่มเมฆจะมีจำนวนไฟล์หรือวัตถุเก็บข้อมูลอยู่เป็นจำนวนมาก ซึ่งการประมวลผลด้วยแบบจำลองอาจจะใช้เวลานาน ทางผู้วิจัยเสนอให้เลือกกลุ่มไฟล์หรือวัตถุเก็บข้อมูลที่มีความนิยมในการดาวน์โหลดหรือการใช้งานสูง ผ่านกระบวนการในแบบจำลองการทำสำเนาและการจัดวางวัตถุแบบพลวัตร์ โดยไม่จำเป็นต้องนำไฟล์ทั้งหมดในระบบจัดเก็บข้อมูลแบบกลุ่มเมฆมาผ่านกระบวนการ เนื่องจากการทำสำเนาและการจัดวางแบบพลวัตร์ในรูปแบบที่งานวิจัยนี้ได้นำเสนอ นั้น เราจะได้ผลจากการกระจายตัวของภาระงาน ซึ่งไฟล์หรือวัตถุเก็บข้อมูลที่จะมีผลมากกับการกระจายตัวของภาระงานนั้นจะเป็นไฟล์ที่มีการใช้งานสูงอยู่แล้ว ถ้าหากไฟล์ที่มีความนิยมหรือมีภาระงานสูงได้รับการทำสำเนาและการจัดวางวัตถุแบบพลวัตร์ที่เหมาะสม ระบบรวมของการจัดเก็บข้อมูลแบบกลุ่มเมฆก็จะมีภาระงานที่ดีตามไปด้วย

เพื่อเป็นการพัฒนาต่อยอดของแบบจำลองเรื่องการทำสำเนาวัตถุแบบพลวัตร์ ถ้าหากสามารถสร้างระบบให้สามารถกำหนดค่าตัวแปร 3 ตัว คือ ค่าเกณฑ์ในการพิจารณาการทำสำเนาวัตถุข้อมูล ค่าคาบเวลาในการพิจารณาการทำสำเนาข้อมูล และค่าอายุของสำเนาวัตถุเก็บข้อมูล ได้อย่างอัตโนมัติโดยใช้รูปแบบการร้องขอการเข้าถึงวัตถุเก็บข้อมูลมาเป็นปัจจัยในการทำนายค่า 3 ค่าดังกล่าวล่วงหน้าได้จะทำให้ระบบมีความยืดหยุ่นและปรับตัวเองอย่างเหมาะสมได้เอง

เอกสารและสิ่งอ้างอิง

- Al Feel, H.T. and M.H. Khafagy. 2011. OCSS: Ontology Cloud Storage System, pp. 9-13.
In Proceedings of 1st International Symposium of Network Cloud Computing and Applications (NCCA). 21-23 November 2011, Toulouse, France.
- Kai, Z., H. Song, L. Liu, J. Gao and G. Cheng. 2011. Hybrid Genetic Algorithm for Cloud Computing Applications, pp. 182-187. *In Services Computing Conference (APSCC) 2011*. 12-15 December 2011, IEEE Asia-Pacific, Jeju, Korea.
- Xu, K., M. Song, X. Zhang and J. Song. 2009. A Cloud Computing Platform Based on P2P, pp. 427-432. *In Proceedings of IT in Medicine & Education 2009*. 14-16 August 2009, IEEE. Jinan, China.
- Myint, J. and T.T. Naing. 2011. A data placement algorithm with binary weighted tree on PC cluster-based cloud storage system, pp. 315-320. *In Cloud and Service Computing International Conference 2011*. 12-14 December 2011, Hong Kong.
- National Institute of Standards and Technology. 2011. **Inst. Stand. Technol. Spec. Publ. 500-292.**
- Qingsong, W., B. Veeravalli, B. Gong, L. Zeng and D. Feng. 2010. CDRM: A Cost-Effective Dynamic Replication Management Scheme for Cloud Storage Cluster, pp. 188-196. *In Cluster Computing (CLUSTER)*. 20-24 September 2010, IEEE International Conference 2010, Heraklion, Crete, Greece.
- Yusoh, Z.I.M. and M. Tang. 2010. A penalty-based genetic algorithm for the composite SaaS placement problem in the Cloud, pp. 1-8. *In Evolutionary Computation (CEC) 2010*. 18-23 July 2010, IEEE Congress, Barcelona, Spain.

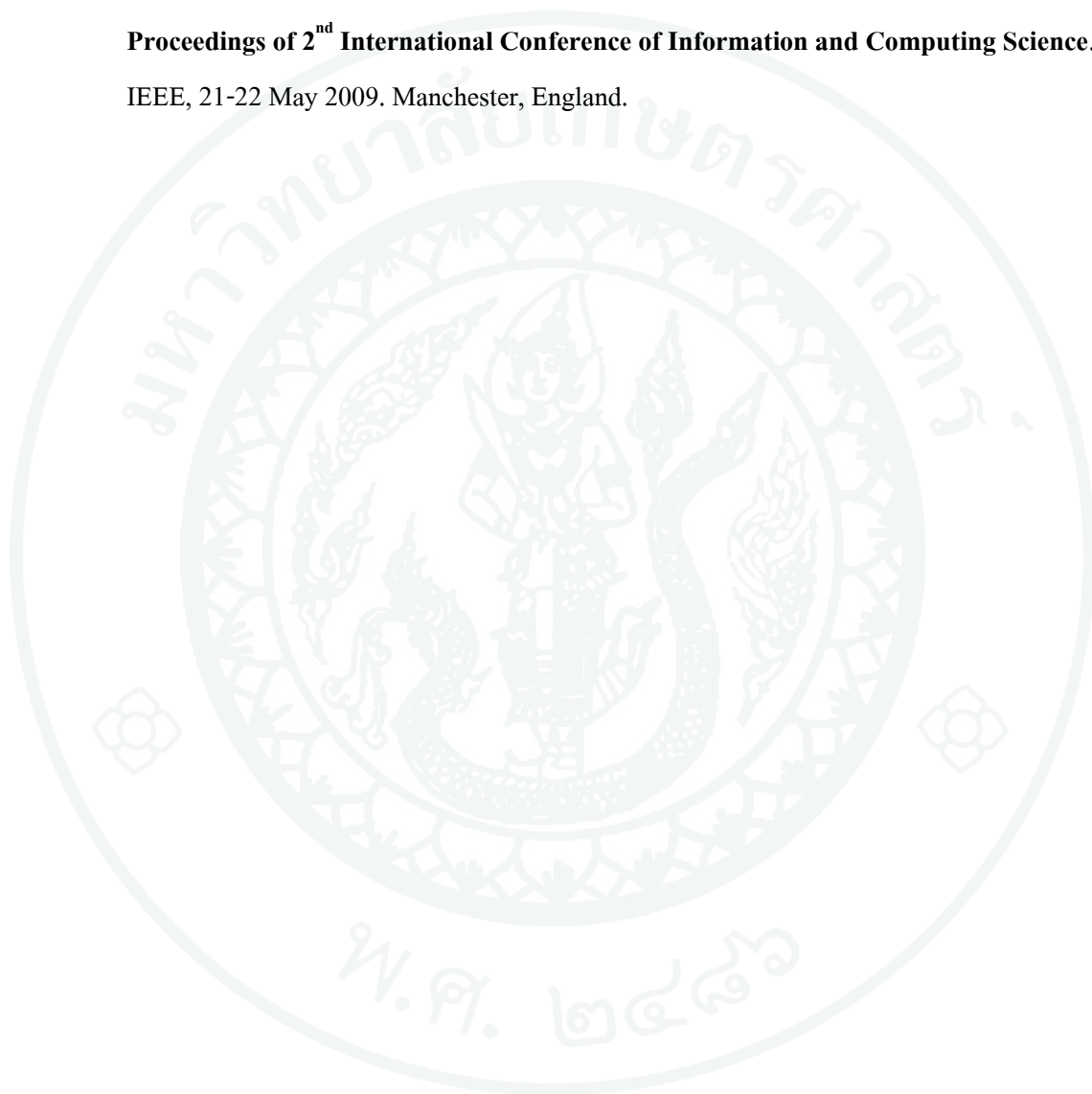
Zeng, Z. and B. Veeravalli. 2008. On the Design of Distributed Object Placement and Load Balancing Strategies in Large-Scale Networked Multimedia Storage Systems.

Knowledge and Data Engineering 20 (3): 369-382.

Zhan, Y. and S. Yong. 2009. Cloud Storage Management Technology, pp. 309-311. *In*

Proceedings of 2nd International Conference of Information and Computing Science.

IEEE, 21-22 May 2009. Manchester, England.



ประวัติการศึกษาและการทำงาน

ชื่อ-นามสกุล	นายกนทร จินดารักษ์
วัน เดือน ปี ที่เกิด	1 มิถุนายน 2528
สถานที่เกิด	อำเภอหาดใหญ่ จังหวัดสงขลา
ประวัติการศึกษา	วศ.บ. (วิศวกรรมศาสตร์สาขาคอมพิวเตอร์) มหาวิทยาลัยสงขลานครินทร์ กรรมการผู้จัดการบริษัท
ตำแหน่งหน้าที่การงานปัจจุบัน	บริษัท สมอลล์ ไลท์ รুম จำกัด
สถานที่ทำงานปัจจุบัน	
ผลงานอื่น	- วิทยากรอบรมแอนดรอย โครงการ MT ² ของซอฟต์แวร์ปาร์ค - วิทยากรอบรมแอนดรอยโครงการ Samart Innovation 2011 - โปรแกรม Droidsans Thai Keyboard สำหรับ ระบบปฏิบัติการแอนดรอย - โปรแกรม ThaiFloods สำหรับระบบปฏิบัติการ แอนดรอย