

การพัฒนาการจัดการภาระงานสำหรับคอมพิวเตอร์เสมือน  
ในระบบประมวลผลแบบกลุ่มเมฆ  
The Development of a Virtual Machine Workload Management  
for a Cloud Computing System

เจษฎากานต์ แสงรัตน์, ภูซงค์ อุทัยภาส  
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์  
b4905092@ku.ac.th, pu@ku.ac.th

ABSTRACT

This paper presents the design and development of software tool for virtual machine workload management for cloud computing system. The goal is to develop software that can control the startup, termination, and movement of virtual machine for VM cloud. More intelligent management such as load balancing can easily be added later. The paper discusses the design concept, present the result from the prototype software developed, and discussed some future work.

**Keywords:** Cloud Computing, Virtual Machine, Scheduler, Cluster

บทคัดย่อ

บทความนี้นำเสนอการออกแบบและพัฒนาเครื่องมือทางซอฟต์แวร์สำหรับบริหารเครื่องเสมือนในระบบประมวลผลแบบกลุ่มเมฆ จุดประสงค์คือการพัฒนาซอฟต์แวร์ที่ควบคุมการเริ่มทำงาน การสิ้นสุดและการเคลื่อนย้ายเครื่องเสมือนในกลุ่มเมฆ การออกแบบทำให้การเพิ่มเติมกลไกที่ชาญฉลาด เช่นการบริหารจัดการเวิร์คโหลดทำได้ง่ายในอนาคต บทความนี้เสนอการออกแบบและผลบางส่วนจากโปรแกรมต้นแบบที่พัฒนาแล้ว และแนวทางพัฒนาในอนาคต

**คำสำคัญ:** การประมวลผลแบบกลุ่มเมฆ, เครื่องเสมือน, ระบบจัดการภาระงาน, คลัสเตอร์

1. บทนำ

ปัจจุบันนี้ความต้องการใช้คอมพิวเตอร์เพิ่มสูงขึ้นทำให้ศูนย์ข้อมูลต่างๆ ต้องดูแลรักษาคอมพิวเตอร์เซิร์ฟเวอร์จำนวนมาก วิธีการแก้ไขปัญหานี้วิธีหนึ่งที่ได้รับคามนิยมในปัจจุบันคือการให้บริการเครื่องเสมือนในรูปแบบการประมวลผลแบบกลุ่มเมฆ (Cloud Computing) ลักษณะการให้บริการคือ ผู้ใช้จะใช้บริการคอมพิวเตอร์เสมือนผ่านเครือข่าย โดยที่ระบบคอมพิวเตอร์เสมือนนี้เป็นเพียงโปรเซสหนึ่งที่ทำงานในเครื่องของผู้ให้บริการเท่านั้น ผู้ใช้สามารถติดตั้งซอฟต์แวร์และปรับตั้งค่าการใช้งานให้กับเครื่องเสมือนของผู้ใช้ได้อย่างอิสระโดยไม่ส่งผลกระทบต่อเครื่องเสมือนของผู้ใช้รายอื่นๆ ในขณะที่ผู้ให้บริการไม่ต้องดูแลซอฟต์แวร์และฮาร์ดแวร์ให้กับเครื่องเสมือนแต่ละเครื่อง ผู้ให้บริการเพียงดูแลระบบให้บริการเครื่องเสมือน และเครื่องเซิร์ฟเวอร์ที่ให้บริการเท่านั้น สาเหตุที่สามารถตั้งระบบลักษณะนี้เกิดมาจากความสามารถในของซอฟต์แวร์ระบบเสมือน (Virtualization) ที่สูงขึ้นมากในระยะหลัง

การให้บริการเครื่องเสมือนแบบกลุ่มเมฆนั้นจำเป็นต้องใช้ทรัพยากรระบบค่อนข้างสูง ผู้ให้บริการจำเป็นต้องใช้คอมพิวเตอร์หลายเครื่องทำงานร่วมกัน เพื่อให้ระบบรองรับความต้องการของผู้ใช้ได้อย่างไรก็ตาม เมื่อระบบมีขนาดใหญ่มากขึ้น ผู้ดูแลระบบอาจดูแลระบบได้ไม่ทั่วถึง ทำให้ความสามารถในการให้บริการลดลง จึงมีผู้พัฒนาซอฟต์แวร์

สำหรับจัดการเครื่องเสมือนในระบบให้บริการลักษณะนี้ขึ้น แต่ซอฟต์แวร์ลักษณะนี้มักเป็นซอฟต์แวร์ที่มีราคาค่อนข้างแพง

งานวิจัยนี้จะพัฒนาระบบจัดการภาระงานสำหรับระบบให้บริการเครื่องเสมือนแบบกลุ่มเมฆ โดยใช้ซอฟต์แวร์โอเพนซอร์สเป็นฐานในการพัฒนา ระบบที่พัฒนาขึ้นทำให้ผู้ดูแลระบบให้บริการเครื่องเสมือนแบบกลุ่มเมฆ สามารถติดตั้งระบบได้ง่าย ควบคุมเครื่องเสมือนในคลัสเตอร์ได้สะดวก ผู้ดูแลระบบสามารถเลือกควบคุมเครื่องเสมือนแต่ละเครื่องให้เปิด ปิด ในโหนดที่ต้องการ และย้ายเครื่องเสมือนระหว่างแต่ละโหนดได้ โดยระบบสามารถทำงานร่วมกับระบบจัดการภาระงาน (Job Scheduler) และไฮเพอร์ไวเซอร์ (Hypervisor) ได้โดยไม่ต้องเขียนโปรแกรมใหม่ หรือเขียนโปรแกรมเพิ่มเติมเพียงเล็กน้อยเฉพาะชุดคำสั่งที่เกี่ยวข้องกับระบบจัดการภาระงาน และไฮเพอร์ไวเซอร์ที่เลือกใช้

## 2. งานวิจัยที่เกี่ยวข้อง

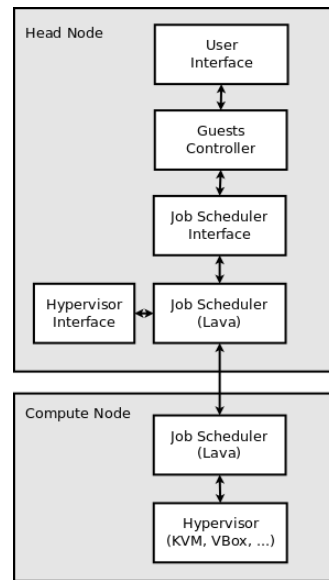
การประมวลผลแบบกลุ่มเมฆ (Cloud Computing) [1] [2] คือรูปแบบการให้บริการทรัพยากรเสมือนแบบออนไลน์ โดยผู้ใช้งานไม่ต้องเป็นเจ้าของ หรือดูแลรักษาทรัพยากรนั้นด้วยตัวเอง รวมถึงสามารถเปลี่ยนลักษณะการใช้งาน และจ่ายค่าบริการแบบจ่ายเท่าที่ใช้ได้ ผู้ให้บริการที่ให้บริการเครื่องเสมือนในรูปแบบการประมวลผลแบบกลุ่มเมฆนี้มีหลายราย เช่น อะเมซอน อีซีทู (Amazon EC2), อีลาสติกโฮสต์ (ElasticHosts) เป็นต้น

ในการตั้งระบบแบบนี้จะอาศัยเทคโนโลยีของ เวอร์ชวลไลเซชัน (Virtualization) [3] คือ การจัดสรรทรัพยากรของระบบด้วยซอฟต์แวร์ เพื่อสร้างสภาพแวดล้อมเสมือนขึ้นมาใหม่ การใช้เครื่องเสมือน จัด เป็น การ ทำ เวอร์ชวลไลเซชันในระดับฮาร์ดแวร์เสมือน (Hardware Abstraction Layer) ซึ่งจำเป็นต้องมีไฮเพอร์ไวเซอร์ (Hypervisor) เป็นซอฟต์แวร์ทำหน้าที่จำลองสถานะเสมือนให้ตรงต่อความเป็นจริง ในปัจจุบันได้มีการพัฒนาระบบสำหรับบริหารคลัสเตอร์เสมือนขึ้นมาบ้าง เช่น VNIX: Managing Virtual Machines on Clusters [4] เป็นระบบจัดการเครื่องเสมือนในคลัสเตอร์ ประกอบด้วยการทำงานร่วมกันของซอฟต์แวร์ศูนย์กลางจัดการเครื่องเสมือน, ซอฟต์แวร์ควบคุมเครื่องเสมือนประจำโหนด, และซอฟต์แวร์ที่ติดตั้งในเครื่องเสมือนของผู้ให้บริการ อีกระบบที่น่าสนใจคือ

STORM: Simple Tool for Resource Management [5] เป็นระบบจัดการเครื่องเสมือนที่ออกแบบเพื่อให้ใช้งานและดูแลรักษาได้ง่าย และใช้โอเอ็มพีไอ (OMPI) ควบคุมการทำงานของระบบเพื่อรองรับการเปลี่ยนแปลงปริมาณการใช้งาน และดูแลระบบให้ใช้พลังงานอย่างประหยัด อย่างไรก็ตามระบบที่ออกแบบมาก่อนยังไม่สามารถประสานงานกับ scheduler ได้ดี ดังนั้นงานนี้จึงมุ่งเน้นการพัฒนาและออกแบบที่สามารถทำงานกับ scheduler ได้ด้วย

## 3. การออกแบบระบบ

ระบบนี้พัฒนาขึ้นด้วยภาษาไพธอน (Python) เวอร์ชัน 2.6 โดยมีสถาปัตยกรรมของระบบแสดงดังภาพที่ 1



รูปที่ 1 สถาปัตยกรรมของระบบ

ระบบแบ่งออกเป็นสามส่วนหลักดังนี้

### 3.1 อินเทอร์เฟซของระบบจัดการภาระงาน

ส่วนนี้เป็นส่วนรวบรวมชุดคำสั่งที่ใช้ติดต่อกับระบบจัดการภาระงานของคลัสเตอร์

### 3.2 อินเทอร์เฟซของไฮเพอร์ไวเซอร์

ส่วนนี้เป็นส่วนรวบรวมชุดคำสั่งที่ใช้ติดต่อกับไฮเพอร์ไวเซอร์เพื่อควบคุมเครื่องเสมือน

### 3.3 โมดูลควบคุมเครื่องเสมือน

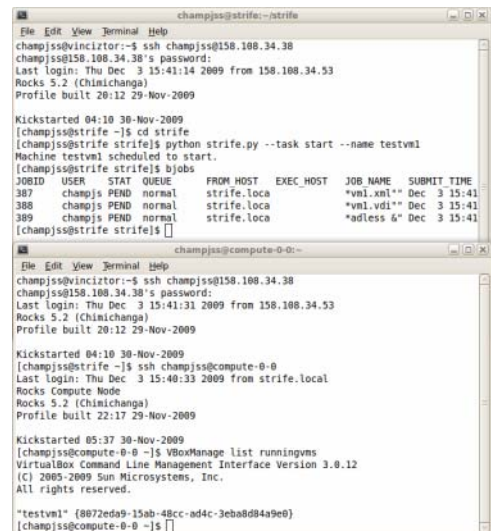
โมดูลนี้เป็นโมดูลหลักของระบบจัดการภาระงาน เมื่อผู้ดูแลระบบตั้งเปิดเครื่องเสมือน โมดูลนี้จะส่งคำสั่งคัดลอกไฟล์ผ่านอินเทอร์เฟซของระบบจัดการภาระงาน เพื่อให้ระบบจัดการภาระงานสั่งให้โหนดเป้าหมายคัดลอกไฟล์จากระบบไฟล์เครือข่าย (NFS) ลงในฮาร์ดดิสก์ของโหนดนั้น จากนั้นจึงเรียกดูคำสั่งเปิดเครื่องเสมือนจากอินเทอร์เฟซของไฮเพอร์ไวเซอร์ แล้วส่งผ่านอินเทอร์เฟซของระบบจัดการภาระงานอีกครั้ง เพื่อสั่งเปิดเครื่องเสมือนนั้น

หากผู้ดูแลระบบตั้งปิดเครื่องเสมือน ระบบจะทำงานสลับลำดับกัน กล่าวคือระบบจะตั้งปิดเครื่องเสมือนก่อน จากนั้นจึงคัดลอกไฟล์เครื่องเสมือนในโหนดนั้นลงในระบบไฟล์เครือข่าย เพื่อปรับปรุงไฟล์เครื่องเสมือนในระบบไฟล์เครือข่ายให้เป็นปัจจุบัน หากผู้ดูแลระบบตั้งเปิดเครื่องเสมือน ระบบจะปิดเครื่องนั้นก่อนแล้วจึงเปิดเครื่องนั้นให้ที่โหนดเป้าหมายใหม่

### 4. ผลการทดลอง

งานวิจัยนี้ทดลองนำระบบจัดการภาระงานที่พัฒนาขึ้นไปติดตั้งในคลัสเตอร์ ซึ่งประกอบด้วยเซดโหนด 1 เครื่องและคอมพิวเตอร์โหนด 3 เครื่อง แต่ละเครื่องมีหน่วยประมวลผลด้วยสัญญาณนาฬิกา 2.4 จิกะเฮิร์ตซ์ 1 หน่วยประมวลผล, หน่วยความจำหลัก (RAM) 2 จิกะไบต์, ฮาร์ดดิสก์ขนาด 90 จิกะไบต์สำหรับเซดโหนด และ 30 จิกะไบต์สำหรับแต่ละคอมพิวเตอร์โหนด แต่ละเครื่องใช้ระบบปฏิบัติการ ร็อกส์ คลัสเตอร์ (Rocks Cluster) เวอร์ชัน 5.2, ระบบจัดการภาระงานลาวา (Lava) เวอร์ชัน 1.0 และใช้เวอร์ชวลบ็อกซ์ (VirtualBox) เวอร์ชัน 3.0.12 เป็นไฮเพอร์ไวเซอร์

ตัวอย่างผลการใช้งานแสดงให้เห็นดังภาพที่ 2



```
champsjs@strife:~$ ssh champsjs@158.108.34.38
champsjs@158.108.34.38's password:
Last login: Thu Dec 3 15:41:14 2009 from 158.108.34.53
Rocks 5.2 (Chimichanga)
Profile built 20:12 29-Nov-2009

Kickstarted 04:10 30-Nov-2009
[champsjs@strife ~]$ cd strife
[champsjs@strife strife]$ python strife.py --task start --name testvm1
Machine testvm1 scheduled to start.
[champsjs@strife strife]$ jobs
JOBID USER START QUEUE FROM HOST EXEC_HOST JOB_NAME SUBMIT TIME
387 champsjs PEND normal strife.local *vml.xml** Dec 3 15:41
388 champsjs PEND normal strife.local *vml.vdi** Dec 3 15:41
389 champsjs PEND normal strife.local *adless 6* Dec 3 15:41
[champsjs@strife strife]$

champsjs@compute-0-0:~$ ssh champsjs@158.108.34.38
champsjs@158.108.34.38's password:
Last login: Thu Dec 3 15:41:31 2009 from 158.108.34.53
Rocks compute Node
Profile built 22:17 29-Nov-2009

Kickstarted 05:37 30-Nov-2009
[champsjs@compute-0-0 ~]$ VBoxManage list runningvms
VirtualBox Command Line Management Interface Version 3.0.12
(C) 2005-2009 Sun Microsystems, Inc.
All rights reserved.

"testvm1" (8072eda9-15ab-48cc-ad4c-3eba8d849e0)
[champsjs@compute-0-0 ~]$
```

รูปที่ 2 แสดงตัวอย่างการใช้งานระบบ โดยหน้าต่างบนแสดงตัวอย่างการเรียกใช้งานระบบ ส่วนหน้าต่างล่างแสดงผลลัพธ์หลังจากเรียกใช้งานระบบแล้ว

จากการทดลองใช้งานพบว่าระบบ ระบบจัดการภาระงานสำหรับระบบให้บริการเครื่องเสมือนแบบกลุ่มเมฆนี้สามารถเปิด ปิด และเคลื่อนย้ายเครื่องเสมือนระหว่างโหนดได้อย่างถูกต้อง นอกจากนี้เนื่องจากระบบทำงานผ่านระบบจัดการภาระงานของคลัสเตอร์อีกทอดหนึ่ง จึงทำให้ไม่เกิดปัญหาโหนดใดโหนดหนึ่งได้รับคำสั่งพร้อมกันมากเกินไปในขณะที่เปิดและปิดเครื่องเสมือน ซึ่งเป็นช่วงที่ต้องใช้ทรัพยากรระบบมาก ผู้ดูแลระบบจึงควบคุมเครื่องเสมือนพร้อมกันได้อย่างต่อเนื่อง รวมถึงใช้งานระบบผ่านสคริปต์ หรือ โปรแกรมอื่นที่เรียกใช้งานระบบนี้ได้อย่างต่อเนื่อง โดยไม่ต้องกังวลถึงปัญหาดังกล่าว

### 5. บทสรุปและข้อเสนอแนะ

อย่างไรก็ตาม งานวิจัยนี้ยังพัฒนาเพิ่มเติมได้อีกมาก เช่น การปรับปรุงระบบให้รองรับการย้ายเครื่องเสมือนโดยไม่ต้องปิดเครื่อง (Live Migration), การปรับปรุงระบบเพื่อให้รองรับการบันทึกสถิติการใช้งานและเก็บค่าบริการ, การเพิ่มความสามารถให้ระบบตัดสินใจเลือกโหนดที่จะให้เครื่องเสมือนไปทำงาน รวมถึงให้ระบบเลือกย้ายเครื่องเสมือนจากโหนดหนึ่งไปยังอีกโหนดหนึ่งได้เอง เพื่อให้ระบบทำงานได้อย่างมีประสิทธิภาพ โดยผู้ดูแลระบบไม่ต้องจัดการส่วนนี้ด้วยตนเอง, การเพิ่ม

ความสามารถให้ผู้ใช้ควบคุมเครื่องเสมือนของผู้ใช้ได้ด้วยตนเองผ่านเว็บ เป็นต้น

## 6. เอกสารอ้างอิง

[1] Gartner Inc., *Gartner Says Cloud Computing Will Be As Influential As E-business*, <http://www.gartner.com/it/page.jsp?id=707508>, 2008

[2] Christian Kane, *Cloud Is Defined, Now Stop the Cloudwashing*, [http://blogs.forrester.com/it\\_infrastructure/cloud-computing/](http://blogs.forrester.com/it_infrastructure/cloud-computing/), 2009

[3] Susanta Nanda, and Tzi-cker Chiueh, *A survey of virtualization technologies*, 2005

[4] Haoyu Tan, Song Wu, Hai Jin Xuan, and Hua Shi, "VNIX: Managing Virtual Machines on Clusters", *Proceedings of the 2008 Japan-China Joint Workshop on Frontier of Computer Science and Technology*, 155-162, 2008

[5] Mark Dehus, and Dirk Grunwald, "STORM: Simple Tool for Resource Management", *Proceedings of the 22nd conference on Large installation system administration conference*, 109-119, 2008