การวัดและเปรียบเทียบประสิทธิภาพ HYPERVISOR ที่ทำงานร่วมกับ HARDWARE ASSISTED VIRTUALIZATION

MEASURING AND COMPARING THE EFFICIENCY OF HYPERVISOR WORKING TOGETHER WITH HARDWARE ASSISTED VIRTUALIZATION

การวัดและเปรียบเทียบประสิทธิภาพ HYPERVISOR ที่ทำงานร่วมกับ HARDWARE ASSISTED VIRTUALIZATION

MEASURING AND COMPARING THE EFFICIENCY OF HYPERVISOR WORKING TOGETHER WITH HARDWARE ASSISTED VIRTUALIZATION

ณัฐกร เฉยศิริ

การศึกษาเฉพาะบุคคลเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต
มหาวิทยาลัยกรุงเทพ
พ.ศ. 2554

©2554
ณัฐกร เฉยศิริ
สงวนลิขสิทธิ์

บัณฑิตวิทยาลัย มหาวิทยาลัยกรุงเทพ
อนุมัติให้การศึกษาเฉพาะบุคคลนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต

เรื่อง การวัดและเปรียบเทียบประสิทธิภาพ Hypervisor ที่ทำงานร่วมกับ Hardware Assisted Virtualization

ผู้วิจัย นาย ณัฐกร เฉยศิริ

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษา [redacted]
(ดร. ธนกร หวังพิพัฒน์วงศ์)

กรรมการผู้ทรงคุณวุฒิ [redacted]
(ดร. ถิรพล์ วงศ์สอาดสกุล)

[redacted]
.....
(ผศ. ดร. ศิวพร หวังพิพัฒน์วงศ์)
คณบดีบัณฑิตวิทยาลัย
วันที่ 1 เดือน กรกฎาคม พ.ศ. 2554

ณัฐกร เฉยศิริ. ปริญญาวิทยาศาสตรมหาบัณฑิต, กรกฎาคม 2554, บัณฑิตวิทยาลัย มหาวิทยาลัยกรุงเทพ.

การวัดและเปรียบเทียบประสิทธิภาพ Hypervisor ที่ทำงานร่วมกับ Hardware Assisted Virtualization (47 หน้า)

อาจารย์ที่ปรึกษา : ดร. ธนกร หวังพิพัฒน์วงศ์

บทคัดย่อ

ปัจจุบันการสร้างเครื่องแม่ข่ายเสมือนด้วยซอฟต์แวร์พัฒนามาจากหลายเทคนิค ทำให้มีประสิทธิภาพและความเหมาะสมกับระบบงานที่แตกต่างกัน การศึกษาวิจัยครั้งนี้มุ่งวัดและเปรียบเทียบประสิทธิภาพของซอฟต์แวร์ที่ใช้ทำเครื่องแม่ข่ายเสมือนโดยใช้เทคนิค Hardware Assisted Virtualization ว่าซอฟต์แวร์ของแต่ละค่ายจะสามารถใช้ประสิทธิภาพในส่วนที่เป็นฮาร์ดแวร์ได้ในอัตราเท่ากันหรือไม่ ด้วยการจำลองสภาพแวดล้อมของเครื่องแม่ข่ายเสมือนและ Workload ที่ใช้ในศูนย์ข้อมูลขึ้นมาทดสอบ แล้ววัดประสิทธิภาพของแต่ละเครื่องแม่ข่ายเสมือนในรูปแบบ Throughput โดยมีเครื่องมือทดสอบเป็น Apache JMeter, PgBench, FFSB และ Iperf ผลการศึกษาพบว่า VMware และ MS Hyper-V มีอัตราการเปลี่ยนแปลงของประสิทธิภาพการทำงานต่างกัน โดย VMware จะมีอัตราการลดลงของประสิทธิภาพต่ำกว่าใน 3 ระบบงานคือ Database Server, Application Server, File Server และมีอัตราการเพิ่มขึ้นของประสิทธิภาพสูงกว่าในระบบงาน Mail Server แต่มีอัตราการลดลงของประสิทธิภาพสูงกว่าในระบบงาน Web Server และพบว่าแต่ละระบบงานด้านไอทีที่มีอัตราการเปลี่ยนแปลงของประสิทธิภาพต่างกัน ซึ่งระบบ Database Server มีอัตราการลดลงของประสิทธิภาพมากที่สุดคือ ใน VMware -5.80% และใน MS Hyper-V -9.40% ส่วนระบบงาน Mail Server มีอัตราการเพิ่มขึ้นของประสิทธิภาพมากที่สุดคือใน VMware 8.25% และใน MS Hyper-V 6.17%

Choeisiri, Nattakorn. Master of Science in Information Technology and Management, July 2011, Graduate School, Bangkok University

Measuring And Comparing The Efficiency Of Hypervisor Working Together With Hardware Assisted Virtualization (47 pp.)

Advisor: Thanakorn Wangpipatwong, Ph.D.

ABSTRACT

Server virtualization software was developed from several techniques which is suitable for each system differently, and result in the different performance. In this research, the evaluation and comparison of the performance of hardware assisted virtualization technique using different hypervisor software with workload in several server environments was conducted. The performance is measured by throughput using Apache JMeter, PgBench, FFSB and Iperf. The result shows that the changing rate of system performance between VMware and MS Hyper-V were different. VMware has decreasing performance rate in 3 servers than MS Hyper-V which are Database Server, Application Server, and File Server, and has increasing performance rate in Mail Server but decreasing performance rate in Web Server. And we found difference changing rate of system performance in each IT workload. Database Server has lower decreasing performance rate in VMware -5.80% , MS Hyper-V -9.40% and Mail Server has higher increasing performance rate in VMware 8.25% , MS Hyper-V 6.17%

กิตติกรรมประกาศ

โครงการศึกษาเฉพาะบุคคล เรื่อง “การวัดและเปรียบเทียบประสิทธิภาพ Hypervisor ที่ทำงานร่วมกับ Hardware Assisted Virtualization” ได้ดำเนินการจนสำเร็จลุล่วง ด้วยความกรุณาจาก ดร.ธนกร หวังพิพัฒน์วงศ์ อาจารย์ที่ปรึกษาโครงการ ที่ได้ให้ความกรุณาเสียสละเวลา ชี้แนะให้คำแนะนำ และแก้ไขจนโครงการศึกษาเฉพาะบุคคลเรื่องนี้เสร็จสมบรูณ์ตามเนื้อหาทางวิชาการและมาตรฐานของการวิจัยเชิงคุณภาพ ผู้วิจัยขอขอบคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอขอบคุณ ดร.ถิรพล วงศ์สอาดสกุล กรรมการผู้ทรงคุณวุฒิ ที่ได้ให้ข้อซักถาม แนะนำที่เป็นประโยชน์ต่อการศึกษาโครงการ ขอบคุณ อาจารย์นัฐธวัช จำปาเงิน ผู้ให้คำปรึกษาเรื่องการติดตั้งระบบสำหรับทดสอบเพื่อใช้ศึกษาค้นคว้าในโครงการ

ขอขอบคุณคณะผู้บริหารมหาวิทยาลัยกรุงเทพ ที่ได้อนุมัติทุนการศึกษาให้ผู้วิจัยได้ศึกษาในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศและการจัดการ

ขอขอบคุณ บิดามารดา ที่ให้การสนับสนุนและเป็นกำลังใจเสมอมา รวมถึงอาจารย์ทุกท่านที่ได้ถ่ายทอดวิชาความรู้ ทั้งในด้านการศึกษาและการดำเนินชีวิต

สุดท้ายนี้ หากผลการศึกษาวิจัยนี้เกิดประโยชน์กับบุคคลที่ได้นำไปใช้งานก็ดี ศึกษาก็ดี หรือเกิดคุณใด ๆ ก็ดี ขอให้คุณงามความดีนั้นสรรเสริญไปยังผู้มีพระคุณของข้าพเจ้า

ณัฐกร เฉยศิริ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ง
บทคัดย่อภาษาอังกฤษ	จ
กิตติกรรมประกาศ	ฉ
สารบัญตาราง	ฌ
สารบัญภาพ	ญ
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการศึกษา	3
1.3 ประโยชน์ที่คาดว่าจะได้รับ	3
1.4 ขอบเขตการศึกษา	4
1.5 ขั้นตอนการศึกษา	4
1.6 ระยะเวลาการศึกษาและแผนงาน	5
บทที่ 2 การทบทวนวรรณกรรม	
2.1 Server Virtualization	6
2.2 Virtualization เทคนิคบนสถาปัตยกรรม x86	6
2.3 ปัญหาของการทำ Virtualization บนสถาปัตยกรรม x86	9
2.4 เทคนิคการสร้างเครื่องแม่ข่ายเสมือนแบบ Hypervisor	10
2.5 การวัดประสิทธิภาพของ Hypervisor	13
2.6 สมมติฐานการวิจัย	16
บทที่ 3 ขั้นตอนการศึกษา	
3.1 จัดเตรียมเครื่องมือในการวัดประสิทธิภาพของ Virtualization	17
3.2 ทดสอบประสิทธิภาพของ VMWare	27
3.3 ทดสอบประสิทธิภาพของ MS Hyper-V	27
3.4 เปรียบเทียบและสรุปผล	27
บทที่ 4 ผลการศึกษา	
4.1 ผลการวัดประสิทธิภาพของแต่ละระบบงาน	29
4.2 ผลการเปรียบเทียบประสิทธิภาพของ Hypervisor	36

สารบัญ(ต่อ)

	หน้า
บทที่ 5 บทสรุปและข้อเสนอแนะ	
5.1 วัตถุประสงค์	40
5.2 สมมติฐานการวิจัย	40
5.3 กระบวนการศึกษา	40
5.4 อภิปรายผลการศึกษาค้นคว้า	41
5.5 ข้อเสนอแนะเพื่อการนำไปใช้	41
5.6 ข้อเสนอแนะเพื่อการวิจัยในอนาคต	41
บรรณานุกรม	43
ประวัติผู้เขียน	47

BANGKOK UNIVERSITY FOUNDED 1962

สารบัญตาราง

		หน้า
ตารางที่ 1	ตารางแสดงอันดับความสำคัญของเทคโนโลยีที่สำรวจโดย Gartne	2
ตารางที่ 2	ตารางแสดงแผนการศึกษาและดำเนินงานในแต่ละขั้นตอน	5
ตารางที่ 3	ตารางแสดงระบบงานและเครื่องมือวัดประสิทธิภาพของแต่ละระบบงาน	21
ตารางที่ 4	ตารางแสดงรายละเอียดการติดตั้งเครื่องแม่ข่ายเสมือนและ Workload ที่ใช้ในการทดสอบ	26
ตารางที่ 5	ตารางบันทึกผลผลการทดลองระบบงาน Web Server	29
ตารางที่ 6	ตารางบันทึกผลผลการทดลองระบบงาน Mail Server	30
ตารางที่ 7	ตารางบันทึกผลผลการทดลองระบบงาน Application Server	31
ตารางที่ 8	ตารางบันทึกผลผลการทดลองระบบงาน Database Server	31
ตารางที่ 9	ตารางบันทึกผลผลการทดลองระบบงาน File Server	33
ตารางที่ 10	ตารางบันทึกผลผลการทดลองระบบงาน Idle Server	34
ตารางที่ 11	ตารางเปรียบเทียบผลการวัดประสิทธิภาพแยกตามระบบงานของ VMWare	37
ตารางที่ 12	ตารางเปรียบเทียบผลการวัดประสิทธิภาพแยกตามระบบงานของ MS Hyper-V	37

สารบัญภาพ

		หน้า
ภาพที่ 1	Hosted Architecture	7
ภาพที่ 2	Hypervisor Architecture	8
ภาพที่ 3	แสดงส่วนประกอบของ Hypervisor	9
ภาพที่ 4	แสดงการเรียกใช้งานชุดคำสั่งพิเศษบน CPU ที่มีเทคโนโลยี Virtualization	12
ภาพที่ 5	แสดงการวางตำแหน่งการทำงานบน Ring ต่างๆ บนสถาปัตยกรรม x86	13
ภาพที่ 6	แสดง Consolidation Stack Unit	14
ภาพที่ 7	แสดง Tile ของ SPEC	16
ภาพที่ 8	สภาพแวดล้อมของการทดสอบโดยใช้ VMware	22
ภาพที่ 9	สภาพแวดล้อมของการทดสอบโดยใช้ MS Hyper-V	22
ภาพที่ 10	แสดงสภาพแวดล้อมที่ใช้ในการทดสอบ	22
ภาพที่ 11	แผนภูมิเปรียบเทียบอัตราการเปลี่ยนแปลงของประสิทธิภาพในแต่ละระบบงาน เมื่อใช้เทคนิค Hardware Assisted Virtualization ของ VMware	38
ภาพที่ 12	แผนภูมิเปรียบเทียบอัตราการเปลี่ยนแปลงของประสิทธิภาพในแต่ละระบบงาน เมื่อใช้เทคนิค Hardware Assisted Virtualization ของ MS Hyper-V	39

บทที่ 1
บทนำ

เอกสารฉบับนี้ เป็นการนำเสนอข้อมูลที่ได้รับในการศึกษาค้นคว้า รวบรวมข้อมูล ทดลองและปฏิบัติจนได้ผลลัพธ์ตามโครงการศึกษาเฉพาะบุคคล 1 และ 2 ซึ่งเป็นส่วนหนึ่งในการศึกษาตามหลักสูตรวิทยาศาสตร์มหาบัณฑิต สาขาเทคโนโลยีสารสนเทศและการจัดการ มหาวิทยาลัยกรุงเทพ

การศึกษาครั้งนี้ เป็นการศึกษาในหัวข้อเรื่อง “การวัดและเปรียบเทียบประสิทธิภาพ Hypervisor ที่ทำงานร่วมกับ Hardware Assisted Virtualization (Measuring and Comparing the Efficiency of Hypervisor Working Together with Hardware Assisted Virtualization)” เพื่อเป็นประโยชน์ให้กับองค์กรต่าง ๆ สามารถนำไปเป็นข้อมูลประกอบในการตัดสินใจเลือกใช้งาน ออกแบบหรือบริหารระบบคอมพิวเตอร์แบบเสมือนได้อย่างมีประสิทธิภาพ และมีความเหมาะสมกับการใช้งานภายในองค์กรของตัวเอง

1.1 ความเป็นมาและความสำคัญของปัญหา

เทคโนโลยี Virtualization เข้ามามีบทบาทสำคัญในการบริหารจัดการโครงสร้างพื้นฐานของระบบไอทีในปัจจุบัน เนื่องจากการทำ Virtualization สามารถช่วยให้องค์กรลดค่าใช้จ่าย (Sundarrajan & Nellitheertha, 2006) เพิ่มเสถียรภาพของระบบ (Farr, Harper, Spainhower, & Xenidis, 2008) ปรับเปลี่ยนโครงสร้างระบบไอทีให้รองรับความต้องการได้ในทุกสภาวการณ์ขององค์กร อีกทั้งยังสะดวกต่อการบำรุงรักษาระบบทั้งที่เป็นซอฟต์แวร์และฮาร์ดแวร์ (Chen & Xin, 2005) ทำให้สามารถใช้ทรัพยากรได้อย่างประสิทธิภาพคุ้มค่ามากขึ้น

จากตารางที่ 1 เป็นรายงานผลสำรวจของ Gartner‘s 2010 ที่สำรวจจาก CIO 1,586 คน พบว่า Virtualization เป็นเทคโนโลยีที่ถูกจัดอยู่ในอันดับความสำคัญสูงที่สุด รองลงมาก็คือ Cloud Computing (Chris, 2010) ซึ่งก็ต้องใช้ เทคโนโลยี Virtualization เป็นพื้นฐานเช่นเดียวกัน

ตารางที่ 1: ตารางแสดงอันดับความสำคัญของเทคโนโลยีที่สำรวจโดย Gartner

Top 10 Business and Technology Priorities in 2010

Top 10 Technology Priorities	Ranking
Virtualization	1
Cloud Computing	2
Web 2.0	3
Networking,Voice and Data Communications	4
Business Intelligence	5
Mobile Techonologies	6
Data/Document Management and Storage	7
Service-oriented applications and architecture	8
Security Technologies	9
IT Management	10

ที่มา: Chris, M. (2010 , January 19). Virtualization and cloud will transform IT. Retrieved by 8 October 2010 from http://www.informationweek.com/blog/main/archives/2010/01/gartner_virtual.html

จะเห็นได้ว่าระบบโครงสร้างพื้นฐานทางไอทีในหลายองค์กร พยายามปรับตัวเข้าหาเทคโนโลยี Virtualization ดังนั้นประเด็นที่ควรจะต้องนำมาพิจารณาในลำดับถัดไปคือ ทำอย่างไรจึงจะสามารถเลือกใช้เทคโนโลยี Virtualization ที่พัฒนามาจากหลากหลายเทคนิค ให้เหมาะสมกับการใช้งานขององค์กร เพื่อที่จะบริหารจัดการทรัพยากรในระบบ Virtualization ที่มีอยู่ให้มีประสิทธิภาพสูงสุดได้

มีงานวิจัยหลายชิ้นที่ศึกษาเพื่อเปรียบเทียบประสิทธิภาพระหว่าง VM ในแต่ละประเภท (Umeno, Paraynot, Teramoto, Kawanot, & Inamasu, 2006) โดยทดสอบไปตามชนิดของ Application (Bhukya & Ramachandram, 2009) (Sundarrajan & Nellitheertha, 2006) เพื่อหาคำตอบว่าเทคโนโลยี Virtualization เหมาะกับงานด้านไอทีทุกประเภทหรือไม่ หรือว่างานไอทีด้านใดสามารถใช้ประโยชน์จาก Virtualization ในแต่ละเทคนิคได้สูงที่สุด แต่เนื่องจากเทคโนโลยี Virtualization มีการพัฒนาอย่างต่อเนื่อง ดังนั้นกระบวนการทดสอบจึงจำเป็นต้องปรับปรุงเพื่อให้ได้ผลลัพธ์ที่ถูกต้องสอดคล้องกับเทคโนโลยีที่เปลี่ยนไปอยู่เสมอ

ประกอบกับในปัจจุบันการสร้างเครื่องแม่ข่ายเสมือนด้วยซอฟต์แวร์บนสถาปัตยกรรม CPU x86 ถูกนำไปใช้งานอย่างกว้างขวางในสภาพแวดล้อมการทำงานจริง (Production Server) ความต้องการทางด้านประสิทธิภาพของเทคโนโลยีนี้จึงมีความสำคัญมาก โดยมีความพยายามที่จะปรับปรุงเทคนิคการทำเครื่องแม่ข่ายเสมือนให้มีประสิทธิภาพมากขึ้น ด้วยการพัฒนาชุดคำสั่งในฮาร์ดแวร์เพิ่มเติม Fisher-Ogden (2006) ได้ศึกษาเรื่อง Hardware Support for Efficient Virtualization พบว่าบริษัทผู้ผลิต CPU ชั้นนำอย่าง Intel และ AMD ได้พัฒนา CPU ที่สนับสนุนเทคนิคการทำเครื่องแม่ข่ายเสมือน ที่เรียกว่า Intel VT-x และ AMD-V โดยมีการเพิ่มชุดคำสั่งพิเศษเพื่อสนับสนุนการทำงานของ Virtual Machine Monitors (VMM) ซึ่งคาดว่าจะช่วยให้มีประสิทธิภาพการทำงานสูงขึ้น

ทางด้านของผู้ผลิตซอฟต์แวร์ที่ใช้ทำเครื่องแม่ข่ายเสมือนก็เริ่มปรับปรุงเทคนิคการทำงาน ให้สามารถเรียกใช้งานชุดคำสั่งพิเศษที่สนับสนุนการทำงานดังกล่าวได้ (VMware Inc, 2009; Rajesh, 2010) โดยที่ได้รับความนิยมสูงในปัจจุบันมีหลายโปรแกรม เช่น VMware, Xen, และ MS Hyper-V ซึ่งแต่ละโปรแกรมถูกพัฒนาโดยใช้เทคนิคต่างกันออกไป จึงมีคุณสมบัติการทำงานที่แตกต่างกัน ส่งผลให้มีประสิทธิภาพการทำงานแตกต่างกันไปอีกด้วย

ดังจะเห็นได้ว่าการพัฒนาเครื่องแม่ข่ายเสมือนมีการพัฒนาคู่กันไปในส่วนที่เป็น “ซอฟต์แวร์” และ “ฮาร์ดแวร์” การศึกษาวิจัยนี้จึงมุ่งที่จะหาคำตอบว่า ซอฟต์แวร์ที่ใช้ทำเครื่องแม่ข่ายเสมือนของแต่ละค่าย จะสามารถใช้ประสิทธิภาพในส่วนที่เป็นฮาร์ดแวร์ได้ในอัตราเท่ากันหรือไม่ เพื่อประโยชน์ในการเลือกใช้งานเทคโนโลยีเครื่องแม่ข่ายเสมือน ที่เหมาะสมกับการใช้งานขององค์กร และสามารถบริหารจัดการทรัพยากรในระบบ Virtualization ที่มีอยู่ให้มีประสิทธิภาพสูงสุดได้ เพราะเป็นที่ทราบกันดีว่าการลงทุนทางด้านโครงสร้างพื้นฐานของระบบไอทีมีค่าใช้จ่ายที่สูงมาก ดังนั้นหากองค์กรเลือกใช้เทคโนโลยีสำหรับโครงสร้างพื้นฐานที่ไม่เหมาะสมกับองค์กรแล้วเกิดความเสียหายทางธุรกิจจนทำให้ต้องเปลี่ยนแปลงระบบจะยิ่งมีค่าใช้จ่ายในการเปลี่ยนแปลงระบบ(Switching Cost) ที่สูงมากกว่า ดังนั้นการเลือกใช้เทคโนโลยีด้านโครงสร้างพื้นฐานจึงเป็นเรื่องที่ต้องตัดสินใจอย่างละเอียดรอบครอบ

1.2 วัตถุประสงค์ของการศึกษา

การศึกษาเรื่อง “การวัดและเปรียบเทียบประสิทธิภาพ Hypervisor ที่ทำงานร่วมกับ Hardware Assisted Virtualization” มุ่งจัดทำเพื่อวัดและเปรียบเทียบประสิทธิภาพของ Hypervisor ว่าเมื่อใช้เทคนิคของ Hardware Assisted Virtualization เข้ามาร่วมทำงานด้วยแล้ว Hypervisor ที่ต่างกันจะมีประสิทธิภาพการทำงานเพิ่มขึ้นในอัตราเท่ากันหรือไม่

1.3 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่คาดว่าจะได้รับจากการดำเนินโครงการศึกษาเฉพาะบุคคลในหัวข้อเรื่อง “การวัดและเปรียบเทียบประสิทธิภาพ Hypervisor ที่ทำงานร่วมกับ Hardware Assisted Virtualization” มีดังนี้

1.3.1 สามารถใช้เป็นข้อพิจารณาในการเลือกใช้เทคโนโลยีชนิดต่าง ๆ ของผู้ผลิตทั้งฮาร์ดแวร์และซอฟต์แวร์ได้อย่างเหมาะสมกับองค์กร

1.3.2 สามารถใช้เป็นแนวทางในการวางระบบโครงสร้างพื้นฐานทางไอทีแบบ Virtualization ได้อย่างเหมาะสมกับความต้องการขององค์กร

1.3.3 ทำให้ทราบว่าประสิทธิภาพที่เพิ่มขึ้นจากเทคนิค Hardware Assisted Virtualization มีผลกับระบบงานไอทีแต่ละประเภทแตกต่างกันหรือไม่

1.4 ขอบเขตการศึกษา

การศึกษาในครั้งนี้จะเป็นการเปรียบเทียบซอฟต์แวร์ประเภท Hypervisor ดังนั้นจึงใช้ Hypervisor ที่เป็นตัวเลือกสำหรับการใช้งานจริงในวงการธุรกิจคือ VMware และ MS Hyper-V มาทดสอบร่วมกับเครื่องแม่ข่ายที่ใช้ CPU ของ Intel รุ่นที่มีเทคโนโลยี Intel VT ซึ่งสนับสนุนการทำงานแบบ Hardware Assisted Virtualization โดยใช้ระบบปฏิบัติการ Redhat Enterprise Linux 5.4 Server เป็น Guest OS เนื่องจากเป็นระบบปฏิบัติการที่ถูกนำไปใช้งานจริงในศูนย์ข้อมูล และใช้ Workload สำหรับการทดสอบประสิทธิภาพแบ่งออกเป็น 6 ระบบงานคือ Web Server, Mail Server, Database Server, Application Server, File Server และ Idle Server

1.5 ขั้นตอนการศึกษา

1.5.1 จัดเตรียมเครื่องมือในการวัดประสิทธิภาพของ Virtualization

1.5.1.1 ศึกษาแนวทางการวัดประสิทธิภาพของ Virtualization

1.5.1.2 ออกแบบการทดลองในการวัด

1.5.1.3 จัดเตรียมข้อมูลที่เป็น Load ในการวัด

1.5.1.4 ติดตั้งและทดสอบวิธีการวัด

1.5.2 ทดสอบประสิทธิภาพของ VMWare

1.5.2.1 ติดตั้ง VMWare และระบบปฏิบัติการ Guest OS

1.5.2.2 ติดตั้ง Application เพื่อใช้เป็น Load ทดสอบ

1.5.2.3 ทดสอบและบันทึกผล

1.5.3 ทดสอบประสิทธิภาพของ MS Hyper-V

1.5.3.1 ติดตั้ง MS Hyper-V และระบบปฏิบัติการ Guest OS

1.5.3.2 ติดตั้ง Application เพื่อใช้เป็น Load ทดสอบ

1.5.3.3 ทดสอบและบันทึกผล

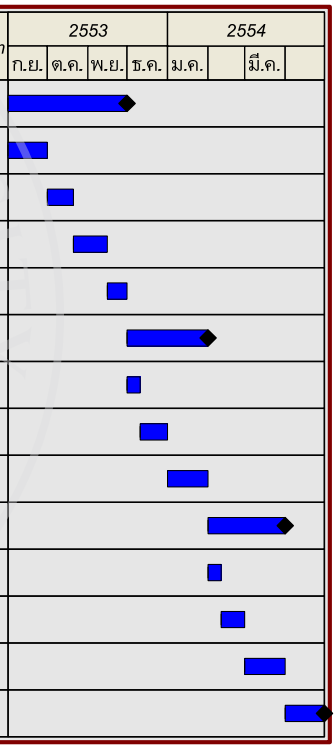
1.5.4 เปรียบเทียบและสรุปผล

1.6 ระยะเวลาการศึกษาและแผนงาน

ระยะเวลาในการศึกษาและการวัดเพื่อทดสอบประสิทธิภาพของซอฟต์แวร์ Hypervisor ภายใต้ระยะเวลาของวิชาโครงการศึกษาเฉพาะบุคคล 1 และ 2

ตารางที่ 2: ตารางแสดงแผนการศึกษาและดำเนินงานในแต่ละขั้นตอน

ID	Task Name	Start	Finish	Duration
1	จัดเตรียมเครื่องมือในการวัดประสิทธิภาพของ Virtualization	1/9/2553	30/11/2553	91d
2	ศึกษาแนวทางการวัดประสิทธิภาพของ Virtualization	1/9/2553	30/9/2553	30d
3	ออกแบบการทดลองในการวัด	1/10/2553	20/10/2553	20d
4	จัดเตรียมข้อมูลที่เป็น Load ในการวัด	21/10/2553	15/11/2553	26d
5	ติดตั้งและทดสอบวิธีการวัด	16/11/2553	30/11/2553	15d
6	ทดสอบประสิทธิภาพของ VMWare	1/12/2553	31/1/2554	62d
7	ติดตั้ง VMWare และระบบปฏิบัติการ Guest OS	1/12/2553	10/12/2553	10d
8	ติดตั้ง Application เพื่อใช้เป็น Load ทดสอบ	11/12/2553	31/12/2553	21d
9	ทดสอบและบันทึกผล	1/1/2554	31/1/2554	31d
10	ทดสอบประสิทธิภาพของ Microsoft Hyper-V	1/2/2554	31/3/2554	59d
11	ติดตั้ง Microsoft Hyper-V และระบบปฏิบัติการ Guest OS	1/2/2554	10/2/2554	10d
12	ติดตั้ง Application เพื่อใช้เป็น Load ทดสอบ	11/2/2554	28/2/2554	18d
13	ทดสอบและบันทึกผล	1/3/2554	31/3/2554	31d
14	เปรียบเทียบและสรุปผล	1/4/2554	30/4/2554	30d



บทที่ 2
การทบทวนวรรณกรรม

2.1 Server Virtualization

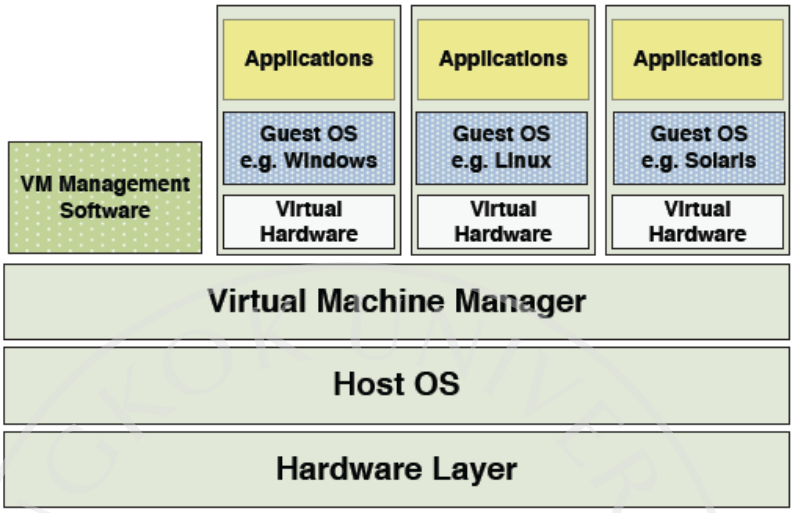
Server Virtualization เกิดมาจากแนวคิดที่ว่าเครื่องแม่ข่ายที่ทำงานแบบ 1 เครื่องต่อ 1 แอพพลิเคชั่นนั้นไม่สามารถใช้งานประสิทธิภาพของเครื่องแม่ข่ายได้อย่างเต็มขีดความสามารถ โดยจากสถิติและงานวิจัยพบว่า เครื่องแม่ข่ายส่วนใหญ่ใช้ประสิทธิภาพในการทำงานจริงเพียงแค่ 15-20% เท่านั้น (Bichler, Setzer, & Speitkamp, 2006) หากสามารถนำเอาทรัพยากรที่ไม่ได้ถูกใช้งานไปใช้งานในด้านอื่นจะได้ประสิทธิภาพเพิ่มขึ้น

นอกจากนี้การที่ CPU และหน่วยความจำของเครื่องแม่ข่ายมีราคาถูกลง แต่มีประสิทธิภาพสูง และมีความจุมากขึ้น จึงเกิดเทคโนโลยีในการทำ Virtualization ด้วยการใช้ทรัพยากรจากเครื่องแม่ข่ายขนาดเล็กตระกูล x86 โดยใช้ซอฟต์แวร์ประเภท Virtualization Management เข้ามาจัดการ และที่ได้รับความนิยมสูงในปัจจุบันก็มีด้วยกันหลากหลายโปรแกรม เช่น VMware, Xen, และ Microsoft Hyper-V ซึ่งแต่ละโปรแกรมได้รับการพัฒนาโดยใช้เทคนิคต่างกันออกไป ทำให้มีคุณสมบัติการทำงานที่แตกต่างกัน ส่งผลให้มีประสิทธิภาพการทำงานที่แตกต่างกันไปอีกด้วย

2.2 Virtualization เทคนิคบนสถาปัตยกรรม x86

ในด้านเทคนิคแล้วการสร้างเทคโนโลยี Virtualization บน CPU ตระกูล x86 สามารถทำได้สองวิธีคือ Hosted Architecture และ Hypervisor Architecture (VMware Inc, 2007)โดยแบบ Hosted นั้นจะ Run Virtualization Layer อยู่บน ระบบปฏิบัติการของเครื่องนั้นเช่นเดียวการ Run Application ปกติดังรูปที่ 1 ซึ่งมีข้อดีก็คือสามารถรองรับฮาร์ดแวร์ได้หลากหลายกว่า

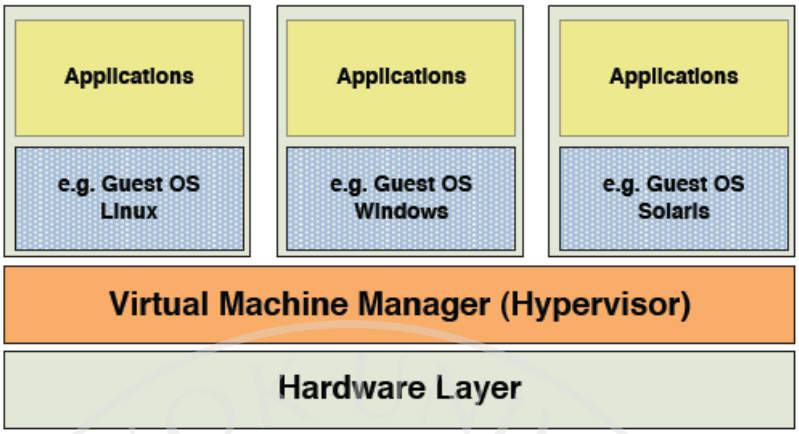
ภาพที่ 1: Hosted Architecture



ที่มา: Marinescu, v. v. (2008, February 29). Design and evaluation of self-management approaches for virtual machine-based environments. Unpublished master's thesis,Wiesbaden University of Applied Sciences,Wiesbaden,Germany.

ในขณะที่แบบ Hypervisor จะ Run Virtualization Layer อยู่บน Hardware ของ x86 ทำให้สามารถเข้าถึงทรัพยากรของฮาร์ดแวร์ได้โดยตรงโดยไม่ต้องผ่านระบบปฏิบัติการอีก จึงทำให้รูปแบบ Hypervisor มีประสิทธิภาพเหนือกว่าแบบ Hosted จึงจะพบว่า ซอฟต์แวร์ที่มักถูกเลือกใช้ในการบริหารจัดการ Virtualization ใน Data Center จะเป็นแบบ Hypervisor ทั้งสิ้น

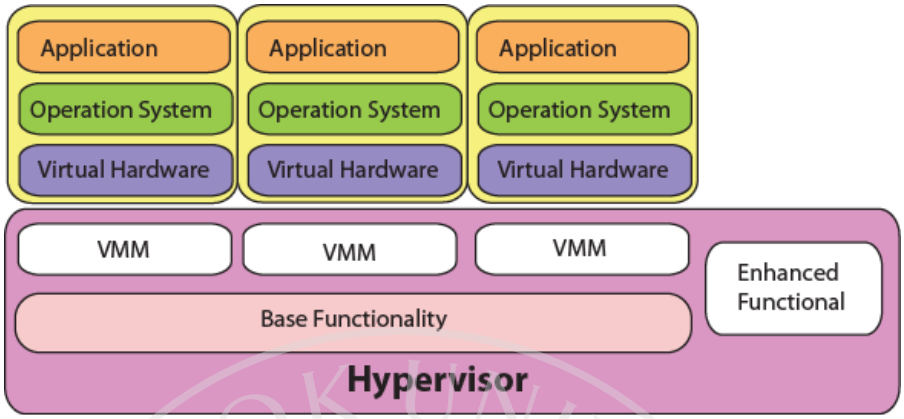
ภาพที่ 2: Hypervisor Architecture



ที่มา: Marinescu, v. v. (2008, February 29). Design and evaluation of self-management approaches for virtual machine-based environments. Unpublished master's thesis,Wiesbaden University of Applied Sciences,Wiesbaden,Germany.

เพื่อให้สามารถเข้าใจการทำงานของ Hypervisor ได้ดีขึ้นขอให้ดูตามภาพประกอบที่ 3 Hypervisor เป็นศัพท์ที่ใช้เรียก Virtualization Layer ซึ่งทำหน้าที่บริหารจัดการ Virtual Machine โดยผ่านทาง Virtual Machine Monitor (VMM) โดย VMM จะถูกสร้างขึ้นภายใน Hypervisor ทำหน้าที่สำหรับสร้าง Virtual Hardware ให้ในแต่ละ Guest OS นอกจากนี้ในแต่ละ VMM ยังแบ่งแยกการทำงานออกจากกันโดยสิ้นเชิง แต่ว่ายังสามารถแบ่งกันใช้ทรัพยากรของ ฮาร์ดแวร์จริงได้

ภาพที่ 3: แสดงส่วนประกอบของ Hypervisor



ที่มา: VMware Inc. (2007). Understanding full virtualization, paravirtualization, and hardware assist. Palo Alto,CA: VMware, Inc

2.3 ปัญหาของการทำ Virtualization บนสถาปัตยกรรม x86

Chen, Hongyi, Li, Zhiying, Nong และ Dan (2003) ได้ศึกษาเรื่องเทคนิคของ Hardware Assisted Full Virtualization พบว่า การสร้างเครื่องแม่ข่ายเสมือนแบบ Hardware Assisted บนสถาปัตยกรรมแบบ x86 มีต้นแบบมาจาก เครื่องแม่ข่ายเสมือนในยุคแรกคือ IBM S/370 ซึ่งใช้หลักการที่ว่า ในทุกๆ คำสั่งที่เป็น Privileged Instruction และถูกเรียกใช้โดย Guest OS จะถูกดักจับไว้ แล้วส่งให้ Virtual machine monitor (VMM) แปลงไปเป็นชุดคำสั่งใหม่ ที่ไม่ส่งผลกระทบกับเสถียรภาพของระบบ หรือของ Guest OS ตัวอื่น ๆ แต่เทคนิคแบบนี้ไม่สามารถนำมาใช้กับสถาปัตยกรรมแบบ x86 ได้ เนื่องจากมีกลไกการทำงานที่ต่างกันออกไป คือจะใช้หลักการของ 2-bit Privilege Level ในการควบคุมการสั่งงาน CPU ซึ่งจะแบ่งได้เป็น 4 Level โดยให้ค่า 0(00) มี Privilege สูงที่สุด และค่า 3(11) มี Privilege ต่ำที่สุด

ระบบปฏิบัติการที่ถูกออกแบบให้ทำงานบนสถาปัตยกรรมแบบ x86 จึงทำงานอยู่ใน 2 รูปแบบคือ

I Kernel Mode ในรูปแบบนี้จะสามารถเรียกใช้คำสั่งใน CPU ได้ทั้งหมดรวมถึง Privileged Instruction ที่ทำงานเกี่ยวข้องกับ I/O Interrupt , Memory Management โดยทำงานอยู่ใน Privilege Level 0 ซึ่งปกติ OS Kernel จะทำงานอยู่ในรูปแบบนี้

II. User Mode ในรูปแบบนี้จะอนุญาตเฉพาะคำสั่งที่จำเป็นสำหรับการประมวลผลข้อมูลโดยทำงานอยู่ใน Privileged Level 3 ซึ่ง Application Program จะทำงานอยู่ในรูปแบบนี้ โดยจะสามารถติดต่อเรียกใช้คำสั่งเกี่ยวกับฮาร์ดแวร์ได้ด้วยการเรียกผ่าน Kernel หรือที่เรียกว่า System Call

ดังนั้นการจะสร้างเครื่องแม่ข่ายเสมือนขึ้นมาจึงจำเป็นจะต้องมีซอฟต์แวร์ Layer ใหม่ที่เรียกว่า VMM หรือ Hypervisor ขึ้นมาซึ่งมีหน้าที่สำคัญคือเป็นผู้ชี้ขาดว่าใครสามารถเข้าไปใช้งานฮาร์ดแวร์ได้โดยไม่ส่งผลกระทบกับ Guest OS ใน Virtual Machine อื่น ๆ และเมื่อจำเป็นต้องมี Hypervisor เข้ามาทำให้ Guest OS ไม่สามารถทำงานอยู่ใน Privilege Level 0 ได้ จึงต้องเลื่อนมาทำงานอยู่ใน Privilege Level อื่นที่สูงกว่า เทคนิคนี้เรียกว่า De-Privileging ซึ่งซอฟต์แวร์ส่วนใหญ่ที่ทำเครื่องแม่ข่ายเสมือนจะใช้เทคนิคนี้ เพราะเป็นการอนุญาตให้ Hypervisor สามารถควบคุมการทำงานของ Guest OS ในการเข้าถึงฮาร์ดแวร์และหลีกเลี่ยงการเข้าไปจองทรัพยากรด้วยตัวเองของ Guest OS

2.4 เทคนิคการสร้างเครื่องแม่ข่ายเสมือนแบบ Hypervisor

การทำเครื่องแม่ข่ายเสมือนแบบ Hypervisor สามารถแบ่งออกได้เป็น 3 เทคนิค (VMware Inc, 2007) ดังต่อไปนี้

2.4.1 Full Virtualization using Binary Translation (BT)

เทคนิคนี้ก็คือการดักจับคำสั่งที่ Kernel ของ Guest OS ต้องการจะเรียกใช้คำสั่งติดต่อกับฮาร์ดแวร์แล้วแปลงให้เป็นชุดคำสั่งที่เหมาะสมต่อสภาพแวดล้อมของ Virtualization คือไม่ครอบครองทรัพยากรระบบไว้เพียง Virtual Machine เดียว นอกจากนี้กระบวนการทำ BT ยังเก็บคำสั่งที่ถูกแปลงแล้วไว้ใน Translator Cache (TC) ซึ่งหมายความว่าหากมีการเรียกใช้คำสั่งซ้ำ ๆ กันเช่นการทำ Loop กระบวนการแปลงจะเกิดขึ้นเพียงครั้งเดียวเท่านั้น สำหรับ Application ที่ Run อยู่ใน User Mode ไม่จำเป็นต้องถูกทำ BT เพราะเป็นชุดคำสั่งที่มี Privilege ระดับต่ำ ดังนั้นจึงสามารถทำคำสั่งได้โดยตรงเหมือนกับทำงานใน Native Mode (Adams & Agesen, 2006)

เทคนิคนี้มีข้อดีคือ ไม่ต้องเปลี่ยนแปลงโครงสร้างภายในของระบบปฏิบัติการ ทำให้การโอนย้าย Virtual Machine ระหว่างฮาร์ดแวร์จริงทำได้ง่าย แต่อย่างไรก็ตามแม้ BT จะถูกออกแบบการทำงานให้มี Overhead น้อยที่สุดแต่กับคำสั่งบางประเภทเช่น System Calls, การ Access คำสั่งใน Chipset และ I/O, Memory Management จะเป็นคำสั่งที่ต้องใช้ Overhead สูงในการแปลง สำหรับโปรแกรมที่ใช้เทคโนโลยีแบบนี้ได้แก่ VMware Workstation และ MS Hyper-V

2.4.2 OS Assisted Virtualization หรือ Para-Virtualization

เป็นเทคนิคที่ไม่ได้แตกต่างจาก BT มากนัก BT เป็นการแปลงคำสั่งที่มี Privilege สูงในขณะที่กำลังจะส่งไปให้ CPU ทำงาน Para-Virtualization ก็ทำแบบนั้นเช่นกัน แต่ทำที่ตัว Source Code ใน Kernel ของ Guest OS เพราะว่าการปรับเปลี่ยนที่ Source code นั้นทำได้ง่ายกว่าการแปลงขณะทำงานจริง และไม่ต้องมี Overhead ในการแปลงอีกด้วย นอกจากนี้ยังตัดการดักจับ

คำสั่งที่ไม่จำเป็นออกจากการทำงานของ Hypervisor ซึ่งจะต้องทำอยู่ตลอดเวลาใน BT แต่ Hypervisor ที่ใช้เทคนิคนี้จะต้องเพิ่ม Hypercall Interfaces เพื่อรองรับการเรียกใช้งานจาก Kernel ของ Guest OS โดยตรงได้ (Lu, Makhlis, & Chen, 2006)

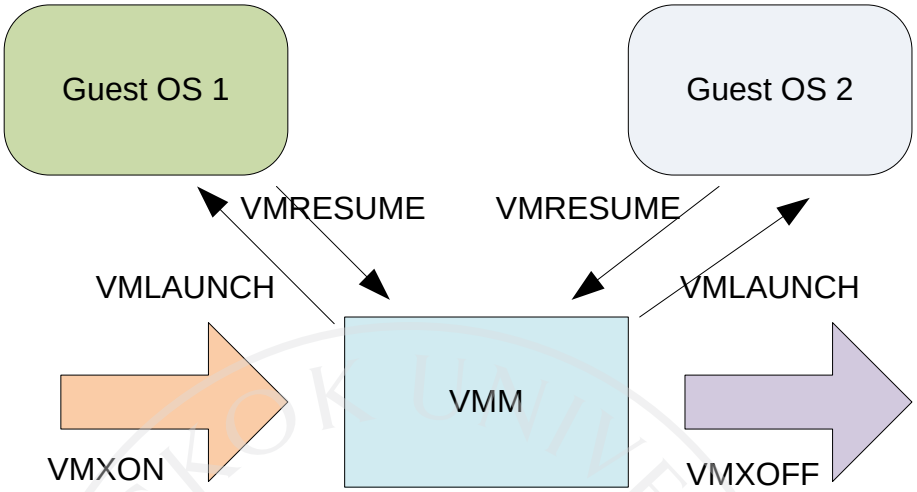
จุดเด่นที่สุดของเทคนิคแบบนี้คือเรื่องของการจัดการ Device Driver เพราะทำให้ Guest OS เสมือนติดต่อกับอุปกรณ์ต่าง ๆ ได้โดยตรงไม่ต้องผ่านการแปลงจากอุปกรณ์เสมือนที่จำลองขึ้นมา ทำให้ได้ประสิทธิภาพใกล้เคียงการทำงานแบบปกติ แต่ว่าข้อเสียของเทคนิคแบบนี้คือใช้ได้เฉพาะระบบปฏิบัติการที่สามารถดัดแปลง Kernel Code ให้รองรับการทำงานแบบ Para-Virtualization ได้เท่านั้น โปรแกรมที่ใช้เทคโนโลยีแบบนี้ได้แก่ Xen ส่วน VMWare ได้นำมาประยุกต์ใช้บางส่วนด้วยการติดตั้งเป็น Device Driver เสริมในชุดโปรแกรม VMWare Tools

2.4.3 Hardware Assisted Virtualization

เทคนิคนี้ไม่ได้ถูกออกแบบขึ้นมาเพื่อ ปรับปรุงการทำงานของ BT หรือ Para-Virtualization แต่ออกแบบขึ้นมาเพื่อแก้ปัญหาของสถาปัตยกรรม x86 ที่แต่เดิมไม่ได้ถูกออกแบบมาไว้ให้รองรับการทำ Virtualized นั่นหมายความว่า เทคนิคนี้ถูกสร้างขึ้นมาด้วยหลักการที่ปรับปรุงมาจาก IBM S/370 คือการดักจับ Privileged Instructions แล้วส่งคำสั่งเหล่านั้นไปให้ Hypervisor ทำงาน

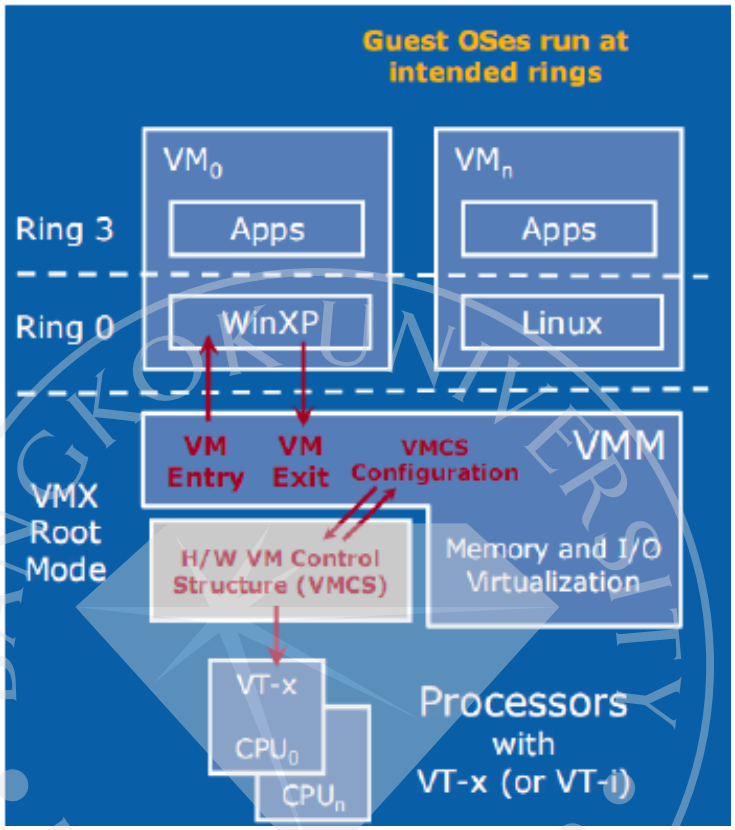
หลักการทำงานของเทคนิคนี้จำเป็นต้องอาศัยการทำงานร่วมกับ CPU ที่มีเทคโนโลยี Virtualization เช่น Intel VT-x และ AMD-V โดย CPU เหล่านี้จะมีชุดคำสั่งพิเศษที่เรียกว่า Virtual Machine Extensions หรือ VMX ซึ่งประกอบไปด้วยชุดคำสั่งที่ใช้สำหรับการทำ Virtualization บน CPU โดยเฉพาะ ได้แก่ VMPTRLD, VMPTRST, VMCLEAR, VMREAD, VMWRITE, VMCALL, VMLAUNCH, VMRESUME, VMOFF, และ VMXON เมื่อ Guest OS ต้องการที่จะเรียกใช้ System Call ที่จำเป็นต้องเข้าสู่กระบวนการทำ Virtualization จะเรียกใช้คำสั่ง VMXON ต่อจากนั้น VMM ก็จะสามารถทำงานบน Virtual Machine แต่ละตัวโดยใช้คำสั่ง VMLAUNCH และออกจาก Virtual Machine ด้วยคำสั่ง VMRESUME เมื่อเสร็จสิ้นการทำงานแล้ว VMM ต้องการปิดและออกจากกระบวนการทำ Virtualization ก็จะเรียกใช้คำสั่ง VMXOFF

ภาพที่ 4: แสดงการเรียกใช้งานชุดคำสั่งพิเศษบน CPU ที่มีเทคโนโลยี Virtualization



ประโยชน์ที่มากที่สุดจากการทำ Hardware Assisted Virtualization ก็คือการที่ Guest OS สามารถกลับไปทำงานใน Privilege Level 0 ได้ตามปกติ ส่วน Hypervisor จะถูกวางไว้ให้ทำงานใน Level ใหม่ที่มี Privilege ที่สูงกว่า เป็น Level ที่ -1 หรือเรียกว่า Root Mode ทำให้ Hypervisor ไม่ต้องแทรกการทำงานแบบอัตโนมัติในทุก ๆ System Call ยกเว้นแต่ System Call เป็นคำสั่งที่มี Privilege สูง (Neiger, Santoni, Leung, Rodgers, & Uhlig, 2006)

ภาพที่ 5: แสดงการวางตำแหน่งการทำงานบน Ring ต่าง ๆ บนสถาปัตยกรรม x86



ที่มา: Johan,D.G.(2008). Hardware accelerated virtualization: Intel VT-x and AMD. Retrieved by 8 March 2010 from http://www.anandtech.com/show/2480/9

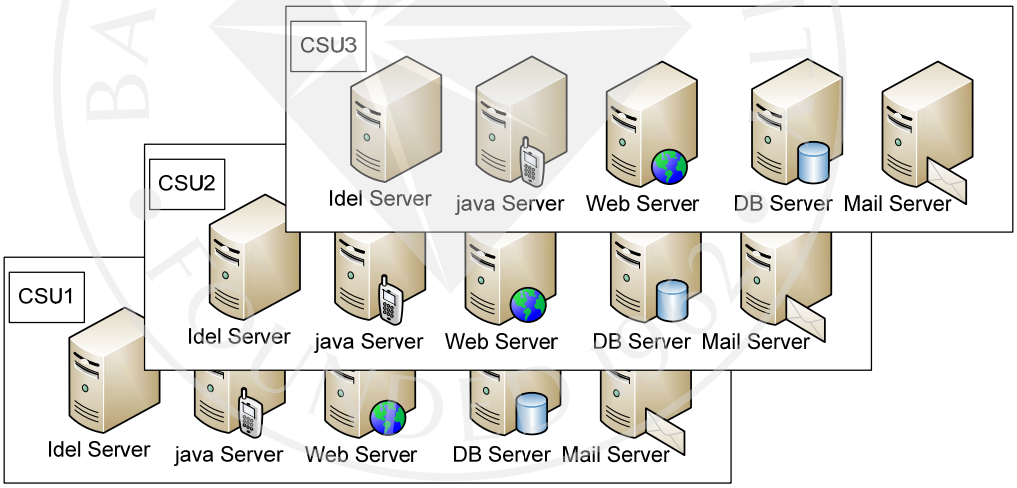
2.5 การวัดประสิทธิภาพของ Hypervisor

Georges และ Eeckhout (2010) ได้ศึกษาวิธีการวัดประสิทธิภาพของระบบ Virtualization พบว่า ยังไม่มีซอฟต์แวร์หรือเครื่องมือชนิดใดที่สามารถวัดประสิทธิภาพของ Hypervisor ได้โดยตรง เนื่องมาจากว่า การทำงานในระบบ Virtualization จะมีส่วนที่ทำงานอยู่อย่างน้อยสองงานคือ Guest OS ที่ Run อยู่บน Virtual Machine และตัว Hypervisor เองที่คอยบริหารจัดการ Virtual Machine ดังนั้นซอฟต์แวร์ที่พัฒนาขึ้นเพื่อวัดประสิทธิภาพของ Virtualization ในปัจจุบันจึงทำในลักษณะการวัด Throughput รวมของระบบทั้งหมดโดยมีตัวอย่างดังต่อไปนี้

2.5.1 vConsolidate

เป็นซอฟต์แวร์ที่ใช้วัดประสิทธิภาพของระบบงาน Virtualization ที่พัฒนาโดยบริษัท Intel ถือเป็นซอฟต์แวร์ยุคแรกที่สนับสนุนการวัดประสิทธิภาพของ Virtualization หลักของการวัดประสิทธิภาพด้วยวิธีนี้คือ จำลอง Consolidation Stack Unit (CSU) ขึ้นมาโดยมีลักษณะดังรูปที่ 1 ซึ่งเป็นตัวอย่างของ CSU จำนวน 3 ชุด แต่ละชุดประกอบไปด้วยเครื่องแม่ข่ายเสมือนที่ทำงานในแต่ละประเภทของระบบงานไอที คือ Web Server, Mail Server, Database Server และ Java Server ในแต่ละ Guest OS แยกจากกัน นอกจากนี้ยังต้องมีอีกหนึ่ง Guest OS ที่ไม่ได้ Run ระบบงานใด ๆ เพื่อจำลองเป็น Idle Server

ภาพที่ 6: แสดง Consolidation Stack Unit



การวัดประสิทธิภาพจะเริ่มวัดทีละ 1 CSU โดย Run Workload ของระบบงาน 3 ครั้ง แล้วนำค่ามัธยฐาน (Median) ของผลการวัดที่ได้มาเป็นคะแนน จากนั้นจะเพิ่มระบบงานเข้าไปในระบบทีละ 1 CSU ทำเช่นนี้ไปเรื่อยๆ จนกระทั่ง คะแนนของผลการวัดในครั้งที่ N CSU น้อยกว่าคะแนนของผลการวัดในครั้งที่ N-1 CSU ซึ่งเป็นการแสดงว่าระบบได้ทำงานจนเต็มประสิทธิภาพแล้ว จากนั้นจะนำผลคะแนนครั้งที่ N-1 กับจำนวน CSU ที่ได้ไปคำนวณหาคะแนนในการวัดประสิทธิภาพ (Theurer, Rister, & Dobbelstein, 2008)

2.5.2 VMmark

เป็นซอฟต์แวร์ที่ใช้วัดประสิทธิภาพของระบบงาน Virtualization ที่พัฒนาโดยบริษัท

VMWare ลักษณะการวัดประสิทธิภาพด้วย VMmark ใช้หลักการเดียวกับ vConsolidate คือ สร้างกลุ่มงานย่อยขึ้นมาเรียกว่า Tile ซึ่งประกอบไปด้วย 6 Virtual Machine โดยแบ่งระบบงานเป็น Web Server, Mail Server, Database Server , Java Server, File Server และ Idle Server ซึ่งจะพบว่าต่างจาก vConsolidation ด้วยการเพิ่มระบบงานของ File Server เข้าไปเท่านั้น

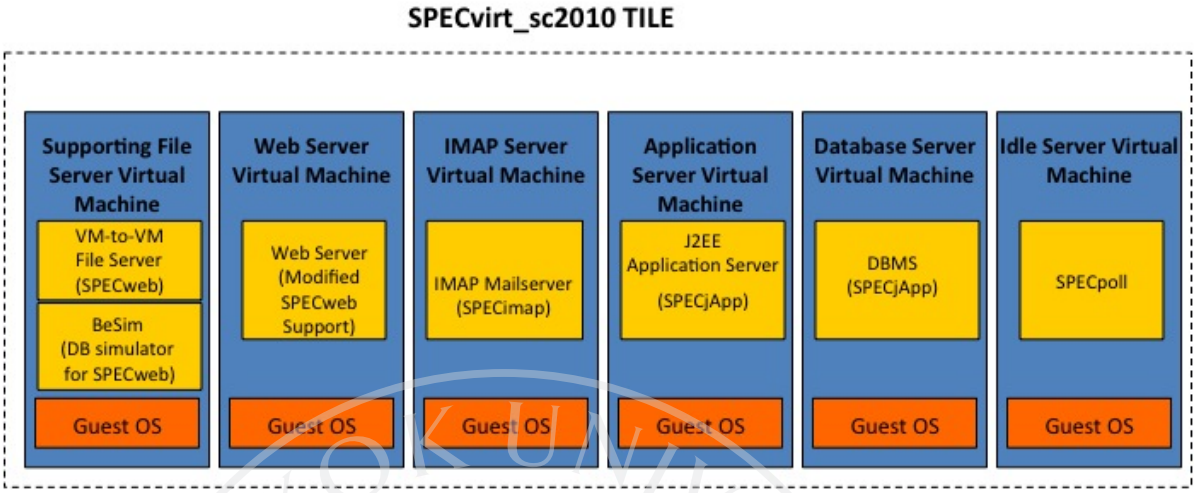
การวัดประสิทธิภาพเริ่มต้นจากการ Run ระบบอย่างน้อย 3 ชั่วโมง เพื่อให้ทุกๆ ระบบงานถูก Run จนเต็มประสิทธิภาพ จากนั้นจะ Run ระบบเพื่อวัดประสิทธิภาพอีก 2 ชั่วโมง โดยในช่วงสองชั่วโมงนี้จะแบ่งบันทึกผลการวัดออกเป็นช่วงย่อยๆ 3 ช่วง ช่วงละ 40 นาที แล้วเลือกค่ามัธยฐานจากคะแนนใน 3 ช่วงนี้เป็นผลลัพธ์ของ Tile สำหรับระบบที่ Run แบบหลาย Tile ก็จะใช้ค่ามัธยฐานของผลรวมในแต่ละ Tile มาเป็นคะแนน (Makhija, Herndon, Smith, Roderick, Zamost, & Anderson, 2006)

2.5.2 SPECvirt

เป็นมาตรฐานการวัดประสิทธิภาพของ Virtualization โดยสมาคม SPEC ซึ่งเกิด จากคณะกรรมการที่เป็นตัวแทนของบริษัทในวงการไอทีหลายๆ บริษัท เช่น AMD, Intel, IBM, HP, VMware, Microsoft โดยมีวัตถุประสงค์เพื่อจะสร้างมาตรฐานของวิธีวัดการประสิทธิภาพบนระบบ Virtualization ซึ่งปัจจุบันพัฒนาจนถึงเวอร์ชั่นที่เรียกว่า SPECvirt_sc2010

SPECvirt_sc2010 ออกแบบการวัดโดยใช้ 3 ระบบงาน คือ Web Server, Java Application Server และ Mail Server ซึ่ง 3 ระบบงานนี้เอาวิธีการวัดมาจากมาตรฐานการวัดระบบงานตัวอื่นๆ ของ SPEC ประกอบไปด้วย SPECweb2005, SPECjAppServer2004 และ SPECmail2008 ที่ได้รับการออกแบบมาจากการสำรวจมาตรฐานระบบงานทั่วๆ ไปที่ใช้กันในศูนย์ข้อมูล ในส่วนวิธีการวัดนั้น SPEC ได้แบ่งกลุ่มของ ระบบงานออกเป็น Tile เช่นเดียวกับ VMmark

ภาพที่ 7: แสดง Tile ของ SPEC



ที่มา: Standard Performance Evaluation Corporation. (2010, June). SPECvirt_sc2010 design document. Retrieved by 2 September 2010 from http://www.spec.org/virt_sc2010/docs/SPECvirt_Design_Overview.html

สำหรับวิธีการวัดประสิทธิภาพก็ใช้หลักการเดียวกับสองวิธีข้างต้นที่กล่าวมาคือ พยายามเพิ่มระบบงานเข้าไปเรื่อยๆ จนกระทั่งระบบทำงานเต็มประสิทธิภาพ แล้ววัดผลตามแต่ละมาตรฐานของระบบงานในแต่ละ Tile (SPECvirt_sc2010 Design Document, 2010)

2.6 สมมติฐานการวิจัย

การที่ผู้ผลิต CPU สร้างชุดคำสั่งพิเศษสำหรับการทำเครื่องแม่ข่ายเสมือนบนสถาปัตยกรรม CPU x86 ที่เรียกว่า Hardware Assisted Virtualization ทำให้ผู้ผลิตซอฟต์แวร์ประเภท Hypervisor นำชุดคำสั่งเหล่านี้ไปใช้งาน ซึ่งถือเป็นการเปลี่ยนเทคนิคการทำเครื่องแม่ข่ายเสมือนจากรูปแบบเดิม ดังนั้นประสิทธิภาพของ Hypervisor ควรจะเปลี่ยนไปในลักษณะที่มีประสิทธิภาพสูงขึ้น แต่ก็อาจจะไม่เสมอไปในทุกๆ Hypervisor และทุกๆ ระบบงานด้านไอที เพราะขึ้นอยู่กับผู้พัฒนาซอฟต์แวร์ว่าจะสามารถพัฒนาให้ใช้ประโยชน์จากเทคนิคนี้ได้มากน้อยเพียงไร ดังนั้นงานวิจัยนี้จึงวัดและเปรียบเทียบประสิทธิภาพของ Hypervisor ที่ใช้เทคนิค Hardware Assisted Virtualization โดยมีสมมุติฐาน 2 ข้อดังนี้

2.6.1 Hypervisor แต่ละชนิด ใช้ประโยชน์จาก Hardware Assisted Virtualization ได้แตกต่างกัน

2.6.2 Hardware Assisted Virtualization ไม่ทำให้ Hypervisor มีประสิทธิภาพสูงขึ้นในทุกสภาพแวดล้อมของระบบงาน

บทที่ 3
ขั้นตอนการศึกษา

ขั้นตอนการศึกษาของโครงการศึกษาเฉพาะบุคคลนี้ ได้ดำเนินการตามแผนที่วางไว้ โดยมีรายละเอียดในแต่ละขั้นตอนดังนี้

3.1 จัดเตรียมเครื่องมือในการวัดประสิทธิภาพของ Virtualization

ในขั้นตอนนี้จะเป็นการค้นคว้าและทดลองเพื่อหาวิธีการที่เหมาะสมในการวัดประสิทธิภาพของระบบ Virtualization

3.1.1 ศึกษาแนวทางการวัดประสิทธิภาพของ Virtualization

ในขั้นตอนนี้จะเป็นการศึกษา เพื่อหาเครื่องมือและศึกษาวิธีใช้เครื่องมือ ในการวัดประสิทธิภาพของ Virtualization โดยยึดหลักการของเครื่องมือวัดประสิทธิภาพที่ได้ค้นคว้ามาในบทที่สองโดย ซึ่งจากการศึกษาค้นคว้าพบว่าวิธีการวัดประสิทธิภาพของ Hypervisor พบว่าในปัจจุบันใช้วิธีวัดจาก Throughput ที่ระบบสามารถทำงานได้ในช่วงเวลาที่กำหนด ดังนั้นจึงกำหนดระบบงานที่จะใช้ทดสอบประสิทธิภาพของระบบตามรูปแบบของโปรแกรม VMMark คือ Web Server, Mail Server, Database Server, Application Server, File Server และ Idel Server จากนั้นจึงแบ่งการศึกษาออกเป็นสองด้านคือ ศึกษาเพื่อเลือกซอฟต์แวร์ที่ใช้เป็นตัวแทนของระบบงานต่าง ๆ โดยศึกษาโปรแกรมที่เป็นที่นิยมใช้งานในศูนย์ข้อมูลทั่วไป และมีมาตรฐานของการทำงานเป็นสากล และศึกษาเพื่อเลือกซอฟต์แวร์สำหรับนำมาวัดประสิทธิภาพของระบบงานต่าง ๆ ดังนี้

3.1.2 ศึกษาซอฟต์แวร์สำหรับใช้ให้บริการระบบงานบนเครื่องแม่ข่าย

3.1.2.1 Apache Web Server

โปรแกรม Apache Web Server เป็นซอฟต์แวร์สำหรับให้บริการระบบ Web Server ที่พัฒนาโดย Apache Software Foundation เครื่องแม่ข่ายที่ติดตั้งโปรแกรมนี้จะสามารถเปิดให้บริการพอร์ท 80 (HTTP Port) ให้แก่ผู้ร้องขอโดยสามารถเชื่อมต่อผ่านโปรแกรมประเภทเว็บบราวเซอร์เพื่อเรียกดูข้อมูลบน Web Server ได้ โปรแกรม Apache Web Server ถูกออกแบบให้สามารถทำงานบน Platform ที่หลากหลายและเป็นแบบ Open Source มักถูกติดตั้งมาพร้อมกับระบบ ปฏิบัติการ Linux จากการที่ไม่เสียค่าลิขสิทธิ์ในการใช้งานและตัวโปรแกรมเองมีการพัฒนา

อย่างต่อเนื่อง รองรับรูปแบบของ HTTP โปรโตคอลที่เป็นมาตรฐาน จึงทำให้เป็นซอฟต์แวร์ประเภท Web Server มีผู้นิยมใช้งานมากที่สุดในปัจจุบัน

3.1.2.2 **Sendmail และ Dovecot**

Sendmail เป็นโปรแกรมที่ใช้รับ-ส่ง อีเมล์ที่มีชื่อเสียงมายาวนาน สนับสนุนกระบวนการทำงานด้านการรับ-ส่ง หลายรูปแบบรวมทั้ง Simple Mail Transfer Protocol (SMTP) ที่ใช้สำหรับรับส่งอีเมล์ผ่านระบบอินเทอร์เน็ต เป็นซอฟต์แวร์ประเภท Open Source นิยมใช้กันมากบนระบบปฏิบัติการตระกูล Linix/UNIX

ส่วน Dovecot เป็นซอฟต์แวร์ประเภท Open Source เช่นเดียวกัน ทำหน้าที่เป็น IMAP และ POP3 Server สำหรับระบบปฏิบัติการ Linux/UNIX ถูกพัฒนาขึ้นเพื่อปรับปรุงระบบรักษาความปลอดภัยในการใช้งานระบบอีเมล์ และต้องการให้ผู้ใช้งานสามารถเข้าถึงอีเมล์ของตนเองได้อย่างรวดเร็ว มีความสามารถทำงานร่วมกับ Mailbox ทั้งแบบมาตรฐาน และ Mailbox แบบ Maildir รองรับการทำงานตามมาตรฐาน UW IMAP และ Courier IMAP

ในการศึกษาครั้งนี้จะใช้โปรแกรม Sendmail ทำหน้าที่เป็น Mail Transport Agent (MTA) และใช้ Dovecot ทำหน้าที่ให้บริการ IMAP และ POP3 Server โดยมีรายละเอียดโปรโตคอลที่ใช้งานดังนี้

- SMTP (Simple Mail Transform Protocol) เป็นโปรโตคอลที่ทำหน้าที่หาเส้นทางและรับ-ส่งอีเมล์ระหว่างเครื่องที่ให้บริการ Mail Server ด้วยกันผ่านทางระบบเครือข่าย ทำงานที่ Port หมายเลข 25
- POP3 (Post Office Protocol Version 3) เป็นโปรโตคอลที่ทำหน้าที่ให้บริการเข้าถึงอีเมล์จากผู้ใช้งานผ่านทางอีเมล์ Client ทำงานที่ Port หมายเลข 110 ลักษณะการทำงานเป็นแบบ Offline คือจะ Download อีเมล์ทั้งหมดที่อยู่ใน Mail Server มาเก็บไว้ในเครื่อง Client
- IMAP (Internet Message Access Protocol) เป็นโปรโตคอลที่พัฒนามาจาก POP3 ทำงานที่ Port หมายเลข 143 ลักษณะการทำงานจะแตกต่างจาก POP3 คือ จะให้บริการเข้าถึงจดหมายบนเครื่อง Mail Server แบบ Online

3.1.2.3 PostgreSQL

PostgreSQL คือระบบจัดการฐานข้อมูลเชิงวัตถุ-สัมพันธ์ (Object-Relational Database Management System หรือ ORDBMS) พัฒนามาจาก University of California, Berkeley ตั้งแต่ปี พ.ศ. 2528 โดยมีวิวัฒนาการมาจากระบบจัดการฐานข้อมูล Ingres เป็นฐานข้อมูลที่มีลักษณะสนับสนุนการทำงานแบบ Transactional Database ในระดับองค์กร มีคุณสมบัติที่สำคัญของระบบฐานข้อมูลคือ ACID (Atomicity, Consistency, Isolation, Durability) ครบถ้วนซึ่ง

สนับสนุน foreign keys, joins, views, triggers, และ stored procedures มีชนิดข้อมูลใน SQL92 และ SQL99 ได้แก่ INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, และ TIMESTAMP

นอกจากนี้ยังสามารถทำงานในหลาย Platformได้แก่ Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), และ Windows เป็นซอฟต์แวร์แบบแบบ Open Source โดยใช้ลิขสิทธิ์ BSD ซึ่งหมายความว่าผู้ใช้สามารถนำไปใช้งานได้ฟรี นอกจากนี้ในปัจจุบัน PostgreSQL ไม่อยู่ภายใต้การควบคุมขององค์กรใดแต่มีผู้ร่วมพัฒนาจากทั่วโลก มีการปรับปรุงพัฒนาโปรแกรมอย่างต่อเนื่อง จึงทำให้ PostgreSQL เป็นที่นิยมในภาคธุรกิจ และภาครัฐในปัจจุบัน

3.1.2.4 AppServ (Apache, PHP, MySQL)

AppServ เป็นชุดติดตั้งโปรแกรมเพื่อทำหน้าที่เป็น PHP Application Server ซึ่งประกอบไปด้วยโปรแกรมต่าง ๆ ดังต่อไปนี้

- Apache Web Server สำหรับทำหน้าที่เป็น Web Server
- PHP สำหรับทำหน้าที่เป็นตัวแปรภาษา PHP
- MySQL สำหรับทำหน้าที่เป็น Database Server
- phpMyAdmin สำหรับทำหน้าที่เป็นโปรแกรมบริหารจัดการฐานข้อมูลของ MySQL

PHP เป็นภาษาในลักษณะ Server-Side Script ชนิดหนึ่ง คือมีการทำงานบนฝั่งเครื่องแม่ข่าย มีรูปแบบการเขียนคำสั่งคล้ายกับภาษา Perl หรือภาษา C และสามารถใช้ร่วมงานกันกับภาษา HTML ได้อย่างมีประสิทธิภาพ สามารถที่จะทำงานเกี่ยวกับ Dynamic Web ได้ทุกรูปแบบเหมือนกับ CGI โดยทั่ว ๆ ไป จุดเด่นของ PHP ก็คือสามารถที่จะติดต่อกับโปรแกรมจัดการฐานข้อมูลที่มีอยู่มากมาย เช่น MySQL, PostgreSQL, MS SQL Server นอกจากนี้ยังมีอีกความสามารถที่พิเศษก็คือ สามารถที่จะติดต่อกับบริการต่าง ๆ ผ่านทางโปรโตคอล เช่น IMAP, SNMP, NNTP, POP3, HTTP และยังสามารถติดต่อผ่านทาง Socket ได้อีกด้วย

การที่ภาษา PHP มีความสามารถในการจัดการเกี่ยวกับแบบฟอร์มข้อมูลที่ถูกสร้างมาจากภาษา HTML และสนับสนุนการติดต่อกับโปรแกรมจัดการฐานข้อมูลได้ง่าย จึงทำให้มีผู้นิยมใช้งาน PHP มากขึ้นอย่างรวดเร็ว นอกจากนี้ยังมีคุณสมบัติอื่นที่ทำให้ผู้ใช้มีความพึงพอใจในการใช้งาน เช่น

- สามารถรันบนระบบปฏิบัติการได้หลากหลายเช่น Windows, Unix และ Linux
- รองรับกับการใช้งานโปรแกรมประเภท Web Server ได้หลายชนิด เช่น Apache, IIS

- เป็นซอฟต์แวร์ฟรีสามารถนำไปใช้งานได้โดยไม่ต้องเสียค่าลิขสิทธิ์

3.1.2.5 Network File System (NFS)

Network File System (NFS) ถูกพัฒนาขึ้นโดย บริษัท Sun Microsystems มีวัตถุประสงค์เพื่อทำให้สามารถ Share ระบบ File System บนระบบเครือข่าย ทำให้เครื่องคอมพิวเตอร์ที่อยู่บนเครือข่ายสามารถใช้ระบบ File System ร่วมกันได้ โดยที่ระบบปฏิบัติการของเครื่องเหล่านั้นไม่จำเป็นต้องเป็นระบบปฏิบัติการเดียวกันกับเครื่องแม่ข่ายที่ให้บริการ NFS Server

ปัจจุบัน NFS พัฒนามาเป็น Version 3 ซึ่งเป็นแบบที่ใช้ RPC (Remote Procedure Call) ซึ่งเป็นโปรโตคอลแบบไม่มีสถานะ (Stateless Protocol) หรือหมายถึงเป็นโปรโตคอลที่ไม่มีการ Acknowledge คือเมื่อมีการับ-ส่งข้อมูลกัน จะไม่ต้องรอคอยการตอบรับการดำเนินการใด ๆ จากอีกด้านหนึ่ง

3.1.3 ศึกษาซอฟต์แวร์สำหรับวัดประสิทธิภาพของระบบงาน

3.1.3.1 Apache JMeter

โปรแกรม Apache JMeter พัฒนาโดย Apache Software Foundation เพื่อใช้สำหรับวัดประสิทธิภาพของระบบบริการข้อมูลบนเครื่องแม่ข่าย โดยสามารถจำลองการร้องขอการใช้บริการจากเครื่องแม่ข่ายได้หลายโปรโตคอล เช่น HTTP SMTP POP3 IMAP สามารถกำหนดจำนวนผู้ร้องขอ และอัตราการร้องขอได้ ผลลัพธ์ที่ได้จะแสดงเป็นจำนวนคำสั่งที่เครื่องแม่ข่ายสามารถให้บริการสำเร็จต่อหนึ่งวินาที (Apache Software Foundation, n.d.)

3.1.3.2 PgBench

โปรแกรม PgBench เป็นโปรแกรมที่ใช้ทดสอบประสิทธิภาพการทำงานของฐานข้อมูล PostgreSQL โดยเฉพาะ ซึ่งเป็นการจำลองการทำ Transaction ประเภท Select/Update/Insert สามารถกำหนดจำนวนผู้ร้องขอ และอัตราการร้องขอได้ ผลลัพธ์ที่ได้จะแสดงเป็นจำนวนคำสั่งที่เครื่องแม่ข่ายสามารถให้บริการสำเร็จต่อหนึ่งวินาที (The PostgreSQL Global Development Group, n.d)

3.1.3.3 FFSB

โปรแกรม FFSB หรือ Flexible File System Benchmark เป็นเครื่องมือทดสอบประสิทธิภาพของระบบ File สามารถนำมาใช้ทดสอบประสิทธิภาพการให้บริการ File ระหว่างเครื่องหรือ Network File System(NFS) ได้ โดยโปรแกรมจะทำการสุ่มเขียน/อ่าน แฟ้มข้อมูลในระบบ File สามารถกำหนดจำนวนผู้ร้องขอ ขนาดแฟ้มข้อมูลและช่วงเวลาที่ใช้ในการร้องขอได้ ผลลัพธ์ที่ได้จะแสดงเป็นจำนวนคำสั่งที่เครื่องแม่ข่ายสามารถให้บริการสำเร็จต่อหนึ่งวินาที (Flexible File System Benchmark, n.d)

3.1.3.4 Iperf

โปรแกรม Iperf เป็นโปรแกรมที่พัฒนาขึ้นเพื่อใช้วัดความสามารถในการส่งข้อมูลบนระบบเครือข่ายระหว่างจุดที่ต้องการทดสอบ สามารถกำหนดชนิดของข้อมูลในการรับส่งได้ เช่น TCP UDP สามารถกำหนดขนาดของข้อมูลในการรับ-ส่ง และช่วงเวลาในการทดสอบได้ ผลลัพธ์ที่ได้จะแสดงเป็นความเร็วในการรับ-ส่งข้อมูล มีหน่วยเป็น Mbps (Iperf, n.d.)

ตารางที่ 3: ตารางแสดงระบบงานและเครื่องมือวัดประสิทธิภาพของแต่ละระบบงาน

ระบบงาน	ซอฟต์แวร์ระบบ	เครื่องมือวัดประสิทธิภาพระบบ
Web Server	Apache	Apache JMeter
Mail Server	Sendmail, Dovecot	Apache JMeter
Database Server	PostgresSQL	PgBench
Application Server	Apache+Php+MySQL	Apache JMeter
File Server	Linux NFS	FFSB
Idle Server	-	Iperf

3.1.4 ออกแบบการทดลองในการวัดประสิทธิภาพ

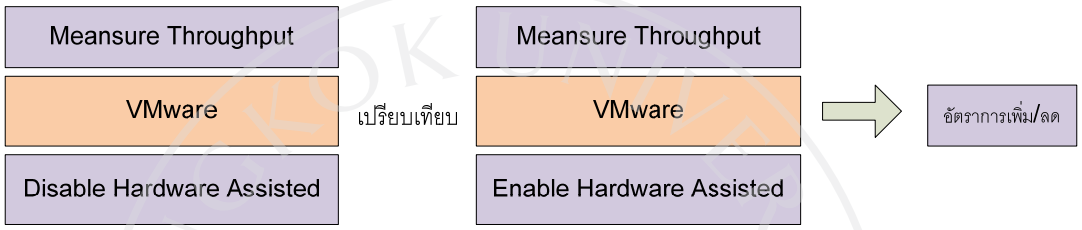
ในขั้นตอนนี้จะเป็นการออกแบบและสร้างสภาพแวดล้อมในการทดสอบ โดยพยายามควบคุมปัจจัยทางด้านสภาพแวดล้อมที่เป็นฮาร์ดแวร์ ซอฟต์แวร์ ในระบบให้เหมือนกันมากที่สุด และจะต้องให้มีความแตกต่างกันเฉพาะ Hypervisor เท่านั้น

การสร้างสภาพแวดล้อมในการทดสอบได้แนวคิดมาจาก Heissler (2010) ซึ่งศึกษาเรื่อง Performance analysis of Xen virtual machines in real-world scenarios และ Bhukya และ Ramachandram (2009) ที่ศึกษาเรื่อง Performance Evaluation of Virtualization and Non Virtualization on Different Workloads using DOE Methodology โดยงานวิจัยดังกล่าววัดประสิทธิภาพของ Hypervisor ด้วยการจำลองสภาพแวดล้อมของเครื่องแม่ข่ายเสมือนและ Workload ที่ใช้ในศูนย์ข้อมูลขึ้นมาทดสอบ

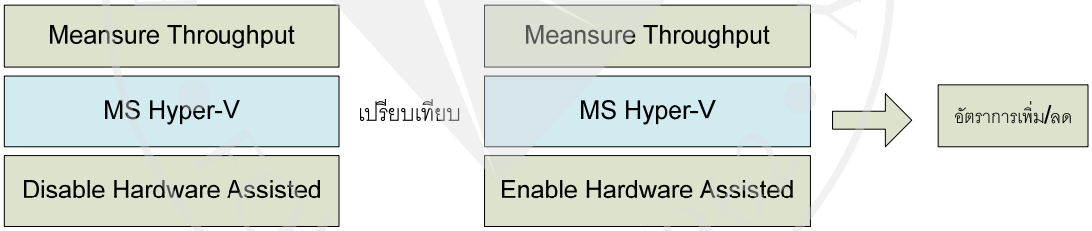
การทดสอบประกอบไปด้วยการจำลองการทำงานของเครื่องแม่ข่ายเปรียบเทียบกันสองระบบ โดยทั้งสองระบบมีสภาพแวดล้อมของ อุปกรณ์ฮาร์ดแวร์ ระบบปฏิบัติการที่เป็น Guest OS และซอฟต์แวร์ที่ให้บริการระบบงานเหมือนกัน แตกต่างกันเพียงใช้ Hypervisor ต่างชนิดกันคือ VMware และ MS Hyper-V จากนั้นจะวัดประสิทธิภาพการทำงานของ Hypervisor แต่ละชนิด

ในสองลักษณะคือเมื่อ Disable และ Enable Hardware Assisted Virtualization ใน CPU โดยการแก้ไขค่า Configuration ในหัวข้อ Intel VT-d Technology และ I/O Virtualization ที่ BIOS ของเครื่องแม่ข่าย แล้วนำผลที่ได้จากการวัดประสิทธิภาพมาเปรียบเทียบหาอัตราการเพิ่มขึ้นหรือลดลงของประสิทธิภาพในการทำงานในแต่ละระบบ ดังจะมีโครงสร้างของการทดลองดังภาพที่ 8 และภาพที่ 9

ภาพที่ 8: สภาพแวดล้อมของการทดสอบโดยใช้ VMware



ภาพที่ 9: สภาพแวดล้อมของการทดสอบโดยใช้ MS Hyper-V



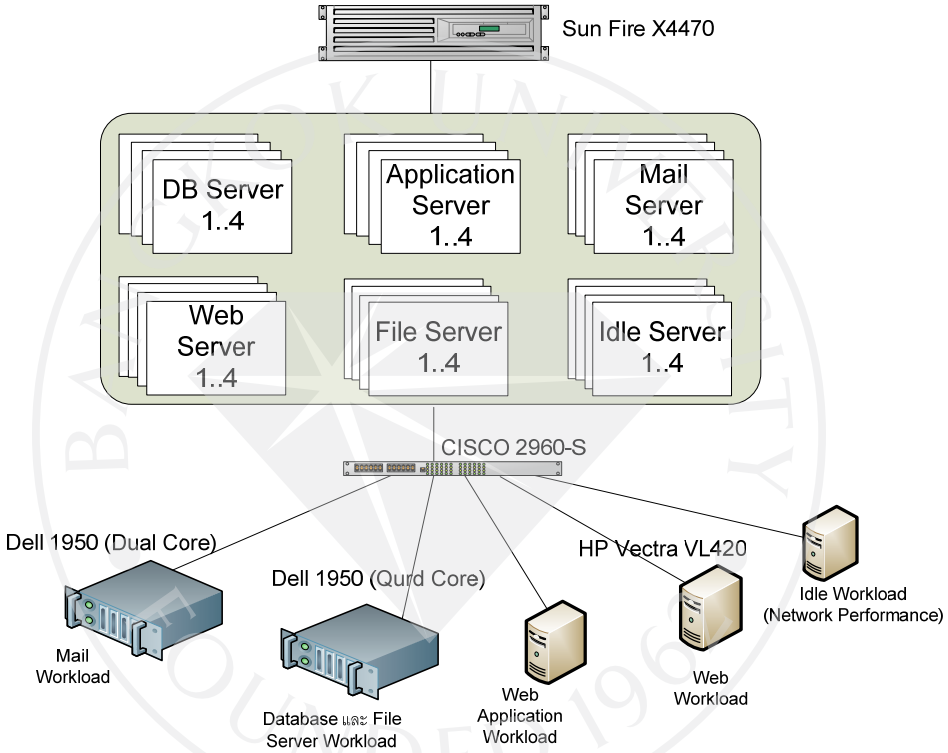
3.1.5 การติดตั้งระบบและทดสอบวิธีการวัด

จากสภาพแวดล้อมในการทดสอบที่ได้ออกแบบไว้ สามารถติดตั้งเครื่องแม่ข่ายเพื่อใช้ทดสอบ โดยใช้หลักการเดียวกับซอฟต์แวร์วัดประสิทธิภาพของระบบ Virtualization ที่ได้ศึกษามา คือสร้างเครื่องแม่ข่ายเสมือนครั้งละ 1 กลุ่ม แล้วติดตั้งโปรแกรมสำหรับให้บริการระบบงานที่มีใช้กันโดยทั่วไปในศูนย์ข้อมูล ซึ่งแบ่งได้ 6 ประเภทคือ Web Server, Mail Server, Database Server, Application Server, File Server และ Idle Server โดยติดตั้ง 1 ระบบงานต่อ 1 เครื่องแม่ข่ายเสมือน จากนั้นทดลองใส่ Workload เข้าไปทดสอบเป็นระยะเวลา 20 นาที หากทุกระบบยังสามารถทำงานได้โดยมีไม่ข้อผิดพลาด จะเพิ่มเครื่องแม่ข่ายเสมือนสำหรับระบบงานเข้าไปทีละกลุ่ม ซึ่งจากการทดสอบพบว่าจำเป็นต้องใช้เครื่องแม่ข่ายเสมือนทั้งหมด 4 กลุ่ม รวมเป็นจำนวนเครื่องแม่ข่ายเสมือนทั้งสิ้น 24 เครื่อง ดังภาพที่ 10 แสดงให้เห็นการแบ่งกลุ่มเครื่องแม่ข่ายเสมือนและแสดง

การเชื่อมต่อของอุปกรณ์ทั้งหมดในสภาพแวดล้อมที่ใช้ทดสอบระบบ

สำหรับระบบปฏิบัติการของเครื่องแม่ข่ายเสมือนใช้ระบบปฏิบัติการเป็น Redhat Enterprise Linux 5.4 Server ใช้ Kernel 2.6.18-164.el5 smp เนื่องจากเป็นระบบปฏิบัติการที่ถูกนำไปใช้งานจริงในศูนย์ข้อมูล และสามารถติดตั้งเครื่องมือในการทดสอบระบบได้ครบถ้วน

ภาพที่ 10: แสดงสภาพแวดล้อมที่ใช้ในการทดสอบ



รายละเอียดเครื่องแม่ข่ายและการติดตั้งระบบงาน

SunFire X4470

Processors :	4 Six-Core Intel Xeon E7530 1.87 GHz
	L1 512 KB / L2 2 MB / L3 12 MB Cache
Memory:	64 GB
Internal Storage:	3x146 GB SAS 10K RPM
Network Interface Card:	1x Broadcom NetXtreme BCM5700 Gigabit Ethernet

รายละเอียดเครื่องลูกข่ายและการติดตั้งเครื่องมือวัดประสิทธิภาพ

Dell 1950 จำนวน 1 เครื่อง

Processors:	1 Dual-Core Intel Xeon E5160 3GHz; 4 MB L2 Cache
Memory:	2 GB
Internal storage:	2x250 GB
Network interface card:	1x Broadcom NetXtreme II BCM5708c Gigabit Ethernet
ระบบปฏิบัติการ (1):	Windows 2000 Server
เครื่องมือวัด:	Apache JMeter
ระบบงานที่วัด:	Mail Server

Dell 1950 จำนวน 1 เครื่อง

Processor:	1 Quad-Core Intel Xeon E5320 1.86 GHz 4 MB L2 Cache
Memory:	4 GB
Internal storage:	2x146 GB
Network interface card:	Broadcom NetXtreme II BCM5708c Gigabit Ethernet
ระบบปฏิบัติการ (2):	Redhat Enterprise Linux 5.4 Server
เครื่องมือวัด:	PgBench และ NFS Service
ระบบงานที่วัด:	Database Server และFile Server

HP VECTRA VL420 จำนวน 3 เครื่อง

Processors:	1 Pentium IV 1.6 GHz
Memory:	256MB
Internal storage:	80 GB
Network interface card:	Intel Pro/100 VE
ระบบปฏิบัติการ (3):	Microsoft Windows XP
เครื่องมือวัด:	Apache JMeter
ระบบงานที่วัด:	Web Server
ระบบปฏิบัติการ (4):	Microsoft Windows XP
เครื่องมือวัด:	Apache JMeter
ระบบงานที่วัด:	Application Server

ระบบปฏิบัติการ (5): Redhat Enterprise Linux 5.4 Server

เครื่องมือวัด: iPref

ระบบงานที่วัด: Idel server

รายละเอียดอุปกรณ์เชื่อมต่อระบบเครือข่าย

Network Switch Cisco Catalyst 2960-S

Port/Speed: 48 ports of Gigabit Ethernet

3.1.6 จัดเตรียมข้อมูลสำหรับเป็น Load ในการวัด

ในขั้นตอนนี้จะเป็นการจำลองการทดสอบเพื่อหา Workload ที่เหมาะสมของแต่ละระบบงาน วิธีการทดสอบกระทำโดยกำหนดระยะเวลาการทดสอบเป็นเวลา 20 นาที ทดสอบด้วยเครื่องแม่ข่ายเสมือน 4 กลุ่ม กลุ่มละ 6 เครื่อง รวมเป็นเครื่องแม่ข่ายเสมือนทั้งหมด 24 เครื่อง ทดสอบด้วยการใส่ Workload ของทุกระบบงานให้กับเครื่องแม่ข่ายเสมือนทุกเครื่องเมื่อครบระยะเวลา 20 นาที จึงตรวจสอบผลการทำงานของแต่ละระบบงานจากเครื่องมือที่ใช้วัดประสิทธิภาพ โดยดูจากผลลัพธ์ที่ได้ว่ามีอัตราการเกิด Error ในระบบงานใดสูงมากหรือไม่ หากพบอัตราการเกิด Error สูงแสดงว่า Workload ที่ใส่ไปในระบบงานนั้นมีปริมาณมากเกินไปจนระบบไม่สามารถทำงานได้ทันจึงควรปรับจำนวน Workload ให้ลดลง แต่ในทางตรงกันข้ามหากมีระบบงานใดไม่มีอัตราการเกิด Error เลย ให้เพิ่มจำนวน Workload ให้กับระบบงานนั้น

ภายหลังจากการทดสอบเพื่อจัดเตรียมข้อมูลสำหรับเป็น Load ในการวัด ทำให้ได้จำนวน Workload ที่เหมาะสมดังตารางที่ 4

ตารางที่ 4: ตารางแสดงรายละเอียดการติดตั้งเครื่องแม่ข่ายเสมือนและ Workload ที่ใช้ในการทดสอบ

ระบบงาน	รายละเอียดเครื่องแม่ข่ายเสมือน	Workload
Web Server	2 vCPU 2 GB Memory 10 GB Hard disk	จำลองการเรียกใช้บริการเว็บเพจ 2 หน้าที่ต่างกัน ด้วย Protocol Http โดยกำหนดให้มีผู้ใช้งานจำนวน 500 คน/หน้า เรียกใช้งานอย่างต่อเนื่องในช่วงเวลา 20 นาที
Mail Server	2 vCPU 2 GB Memory 10 GB Hard disk	จำลองการเรียกใช้งานระบบ Mail 3 ชนิดงาน คือ SMTP POP3 และ IMAP ด้วยผู้ใช้งานจำนวน 200 คน/ชนิดงาน อย่างต่อเนื่องในช่วงเวลา 20 นาที
Database Server	4 vCPU 2 GB Memory 14 GB Hard disk	จำลองการเรียกค้นหา ปรับปรุง และเพิ่มข้อมูลลงในฐานข้อมูล ด้วยจำนวนผู้ใช้ 400 คน อย่างต่อเนื่องในช่วงเวลา 20 นาที
Application Server	2 vCPU 2 GB Memory 10 GB Hard disk	จำลองการเรียกใช้งานหน้าเว็บ Application 2 หน้าที่ต่างกัน ด้วยผู้ใช้งานจำนวน 400 คน/หน้า อย่างต่อเนื่องในช่วงเวลา 20 นาที
File Server	2 vCPU 2 GB Memory 12 GB Hard disk	จำลองการเขียน/อ่าน File ผ่าน Network File System ด้วยผู้ใช้งานจำนวน 400 คนอย่างต่อเนื่องในช่วงเวลา 20 นาที
Idel Server	2 vCPU 2 GB Memory 10 GB Hard disk	วัดประสิทธิภาพการรับ-ส่งข้อมูลบนเครือข่าย โดยทดสอบส่งข้อมูลจากเครื่องลูกข่ายไปยังเครื่องแม่ข่าย แล้ววัดอัตราเร็วเฉลี่ยในการส่งในช่วงเวลา 20 นาที

3.2 ทดสอบประสิทธิภาพของ VMWare

ในขั้นตอนนี้เป็นการทดสอบประสิทธิภาพของระบบเครื่องแม่ข่ายเสมือนโดยการใช้ VMWare Version 4.1 เป็น Hypervisor

3.2.1 ติดตั้งระบบสำหรับทดสอบ

ในขั้นตอนนี้เป็นการติดตั้ง VMWare และระบบปฏิบัติการ Guest OS จากนั้นติดตั้ง Application ของแต่ละระบบงานเพื่อใช้ในทดสอบตามวิธีการที่ได้ศึกษามาในข้อ 3.1

3.2.2 ทดสอบและบันทึกผล

ในขั้นตอนนี้จะเป็นการทดสอบเพื่อหาประสิทธิภาพของระบบ Virtualization โดยเริ่มจากการ Disable Hardware Assisted Virtualization ใน BIOS ของเครื่องแม่ข่าย จากนั้นใส่ Workload ที่ได้กำหนดไว้ของแต่ละระบบงาน เข้าไป Run ทดสอบเป็นระยะเวลา 20 นาที/ครั้ง ทดสอบซ้ำทั้งหมด 6 ครั้ง นำผลการทดสอบเฉพาะครั้งที่ 2-6 มาหาค่าเฉลี่ย โดยแยกคิดเป็นค่าเฉลี่ยของแต่ละระบบงาน แล้วนำมาเป็นคะแนนในการเปรียบเทียบ (ตัดผลการทดสอบครั้งที่ 1 ออกเพื่อหลีกเลี่ยงปัจจัยที่เกิดจาก Cache ในหน่วยความจำและ CPU) เมื่อบันทึกผลการทดลองเรียบร้อยแล้วก็จะปิดระบบทั้งหมดเพื่อไปเปลี่ยนค่า Configuration ใน BIOS ให้ Enable Hardware Assisted Virtualization แล้วดำเนินการทดสอบด้วยกระบวนการเดิมอีกครั้งหนึ่ง

3.3 ทดสอบประสิทธิภาพของ Microsoft Hyper-V

ในขั้นตอนนี้เป็นการทดสอบประสิทธิภาพของระบบเครื่องแม่ข่ายเสมือนโดยการใช้ MS Hyper-V (Windows 2008 R2) เป็น Hypervisor

3.3.1 **ติดตั้งระบบสำหรับทดสอบ**

ในขั้นตอนนี้เป็นการติดตั้ง MS Hyper-V และระบบปฏิบัติการ Guest OS จากนั้นติดตั้ง Application ของแต่ละระบบงานเพื่อใช้ในทดสอบตามวิธีการที่ได้ศึกษามาในข้อ 3.1

3.3.2 **ทดสอบและบันทึกผล**

ในขั้นตอนนี้จะดำเนินการเช่นเดียวกับการทดสอบประสิทธิภาพด้วย VMWare เพื่อควบคุมสภาพแวดล้อมของการทดสอบให้มีความเหมือนกันมากที่สุด

3.4 เปรียบเทียบและสรุปผล

ในขั้นตอนนี้จะเป็นการนำผลการทดลองมาประมวลผลเพื่อหาค่าเฉลี่ยของการวัดประสิทธิภาพของเครื่องแม่ข่ายในแต่ละระบบงานโดยใช้สูตรคำนวณดังนี้

$$\bar{x} = \frac{\sum x}{N}$$

เมื่อ $\bar{x}$ แทนคะแนนเฉลี่ย

$\sum x$ แทนผลรวมของคะแนนทั้งหมด

N แทนจำนวนข้อมูล

เมื่อได้ค่าเฉลี่ยของประสิทธิภาพเครื่องแม่ข่ายในแต่ละระบบงานแล้วจะนำมาคำนวณเทียบเป็นร้อยละอัตราการเปลี่ยนแปลงของประสิทธิภาพในแต่ละระบบงานเมื่อปิดการใช้งาน Hardware Assisted Virtualization เทียบกับเปิดการใช้งาน Hardware Assisted Virtualization โดยทำเป็นตารางเพื่อเปรียบเทียบข้อมูลระหว่าง Hypervisor สองค่ายคือ VMWare และ MS Hyper-V

บทที่ 4
ผลการศึกษา

จากการวัดประสิทธิภาพของ Hypervisor โดยแยกวัดเป็นแบบที่ไม่มีการทำงานร่วมกับ Hardware Assisted Virtualization และแบบที่ทำงานร่วมกับ Hardware Assisted Virtualization พบว่ามีผลการทดสอบเป็นที่น่าสนใจในสองด้านคือ ด้านแรกพบว่า Hypervisor ที่ต่างชนิดกันมีอัตราการเพิ่มขึ้นหรือลดลงของประสิทธิภาพที่แตกต่างกันเมื่อมีการใช้ Hardware Assisted Virtualization ในด้านที่สองพบว่า ในแต่ละระบบงานที่นำมาทดสอบ การเปิดใช้งาน Hardware Assisted Virtualization ไม่ทำให้ Hypervisor มีประสิทธิภาพสูงขึ้นในทุกสภาพแวดล้อมของระบบงาน

4.1 ผลการวัดประสิทธิภาพของแต่ละระบบงาน

ผลการวัดประสิทธิภาพของแต่ละระบบงานมีรายละเอียดดังตารางบันทึกผลการทดลองดังต่อไปนี้

ตารางที่ 5: ตารางบันทึกผลผลการทดลองระบบงาน Web Server

ครั้งที่	VMWare		Hyper-V	
	Disable	Enable	Disable	Enable
ค่าเฉลี่ยครั้งที่ 1	19.80	20.50	22.80	19.70
ค่าเฉลี่ยครั้งที่ 2	24.00	20.60	22.00	20.20
ค่าเฉลี่ยครั้งที่ 3	20.30	20.90	21.10	20.60
ค่าเฉลี่ยครั้งที่ 4	19.80	20.10	20.50	20.30
ค่าเฉลี่ยครั้งที่ 5	20.70	23.10	20.50	20.50
ค่าเฉลี่ยครั้งที่ 6	22.00	19.50	19.70	21.40

จากตารางที่ 5 เป็นผลการวัดประสิทธิภาพของระบบงาน Web Server ที่ได้จากโปรแกรม Apache JMeter ซึ่งโปรแกรมนี้ให้ผลลัพธ์เป็นค่าเฉลี่ยประสิทธิภาพของเครื่องแม่ข่ายเสมือนจำนวน 4 เครื่องต่อระบบงานที่ใช้ทดสอบ โดยทดสอบเป็นจำนวน 6 ครั้ง พบว่าเมื่อ Enable Hardware Assisted Virtualization จะมีประสิทธิภาพการทำงานของระบบลดลงทั้งใน VMWare และ MS Hyper-V

ตารางที่ 6: ตารางบันทึกผลผลการทดลองระบบงาน Mail Server

ครั้งที่	VMWare		Hyper-V	
	Disable	Enable	Disable	Enable
ค่าเฉลี่ยครั้งที่ 1	357.60	361.50	405.30	378.70
ค่าเฉลี่ยครั้งที่ 2	369.30	445.60	467.20	320.60
ค่าเฉลี่ยครั้งที่ 3	212.90	233.50	487.70	290.90
ค่าเฉลี่ยครั้งที่ 4	359.00	483.50	367.20	429.00
ค่าเฉลี่ยครั้งที่ 5	371.50	191.70	302.70	463.30
ค่าเฉลี่ยครั้งที่ 6	375.10	472.70	241.20	477.40

จากตารางที่ 6 เป็นผลการวัดประสิทธิภาพของระบบงาน Mail Server ที่ได้จากโปรแกรม Apache JMeter ซึ่งโปรแกรมนี้ให้ผลลัพธ์เป็นค่าเฉลี่ยประสิทธิภาพของเครื่องแม่ข่ายเสมือนจำนวน 4 เครื่องต่อระบบงานที่ใช้ทดสอบ โดยทดสอบเป็นจำนวน 6 ครั้ง พบว่าเมื่อ Enable Hardware Assisted Virtualization จะมีประสิทธิภาพการทำงานของระบบเพิ่มขึ้นทั้งใน VMWare และ MS Hyper-V

ตารางที่ 7: ตารางบันทึกผลผลการทดลองระบบงาน Application Server

ครั้งที่	VMWare		Hyper-V	
	Disable	Enable	Disable	Enable
ค่าเฉลี่ยครั้งที่ 1	35.10	20.60	4.40	3.90
ค่าเฉลี่ยครั้งที่ 2	20.20	28.90	3.50	3.50
ค่าเฉลี่ยครั้งที่ 3	30.00	25.50	3.60	3.40
ค่าเฉลี่ยครั้งที่ 4	30.80	24.50	3.40	3.40
ค่าเฉลี่ยครั้งที่ 5	31.80	27.50	4.10	3.30
ค่าเฉลี่ยครั้งที่ 6	31.20	31.20	4.10	3.50

จากตารางที่ 7 เป็นผลการวัดประสิทธิภาพของระบบงาน Application Server ที่ได้จากโปรแกรม Apache JMeter ซึ่งโปรแกรมนี้ให้ผลลัพธ์เป็นค่าเฉลี่ยประสิทธิภาพของเครื่องแม่ข่ายเสมือนจำนวน 4 เครื่องต่อระบบงานที่ใช้ทดสอบ โดยทดสอบเป็นจำนวน 6 ครั้ง พบว่าเมื่อ Enable Hardware Assisted Virtualization จะมีประสิทธิภาพการทำงานของระบบลดลงทั้งใน VMWare และ MS Hyper-V

ตารางที่ 8: ตารางบันทึกผลผลการทดลองระบบงาน Database Server

ครั้งที่	เครื่องที่	VMWare		Hyper-V	
		Disable	Enable	Disable	Enable
1	1	118.31	67.77	13.10	9.09
	2	121.51	39.60	12.82	8.72
	3	121.36	41.64	13.19	9.92
	4	118.95	38.21	12.66	9.18

(ตารางมีต่อ)

ตารางที่ 8 (ต่อ): ตารางบันทึกผลผลการทดลองระบบงาน Database Server

ครั้งที่	เครื่องที่	VMWare		Hyper-V	
		Disable	Enable	Disable	Enable
2	1	65.93	56.71	12.82	10.95
	2	58.66	46.62	13.54	9.56
	3	56.84	47.75	13.04	11.24
	4	46.87	44.96	12.79	10.66
3	1	56.16	57.54	13.18	10.93
	2	78.24	57.11	13.32	12.36
	3	82.99	55.48	12.78	11.25
	4	48.95	57.23	12.87	11.68
4	1	69.21	75.04	12.99	11.69
	2	61.10	75.30	13.55	10.97
	3	129.67	71.08	13.43	13.22
	4	61.90	81.91	13.03	12.24
5	1	101.79	95.67	13.62	12.27
	2	68.44	74.78	13.55	11.26
	3	112.71	75.05	13.54	13.97
	4	68.08	90.24	12.68	12.84
6	1	120.02	97.14	14.11	12.90
	2	66.98	95.57	13.90	11.99
	3	104.62	93.58	13.70	14.87
	4	66.49	88.36	13.14	13.74

จากตารางที่ 8 เป็นผลการวัดประสิทธิภาพของระบบงาน Database Server ที่ได้จากโปรแกรม PgBench ซึ่งโปรแกรมนี้ให้ผลลัพธ์เป็นประสิทธิภาพของเครื่องแม่ข่ายจำนวน 4 เครื่องต่อการทดสอบ 1 ครั้ง ซึ่งภายหลังจากการทดสอบเป็นจำนวน 6 ครั้ง พบว่าเมื่อ Enable Hardware Assisted Virtualization จะมีประสิทธิภาพการทำงานของระบบลดลงทั้งใน VMWare และ MS Hyper-V

ตารางที่ 9: ตารางบันทึกผลผลการทดลองระบบงาน File Server

ครั้งที่	เครื่องที่	VMWare		Hyper-V	
		Disable	Enable	Disable	Enable
1	1	638.47	328.33	505.51	547.39
	2	580.91	316.39	420.63	430.60
	3	546.40	333.59	526.08	457.36
	4	520.67	331.66	582.84	444.97
2	1	592.13	503.69	510.61	547.39
	2	537.92	469.70	416.03	430.60
	3	519.41	500.33	534.13	457.36
	4	493.53	474.44	582.99	444.97
3	1	503.69	290.09	508.62	554.55
	2	450.67	504.62	409.51	432.98
	3	483.88	515.25	529.57	462.77
	4	541.01	334.58	576.49	451.43
4	1	643.42	543.83	497.16	542.41
	2	575.67	487.64	423.01	429.30
	3	553.90	483.37	503.14	456.12
	4	543.48	465.80	580.46	423.24

(ตารางมีต่อ)

ตารางที่ 9 (ต่อ): ตารางบันทึกผลผลการทดลองระบบงาน File Server

ครั้งที่	เครื่องที่	VMWare		Hyper-V	
		Disable	Enable	Disable	Enable
5	1	467.57	494.21	475.02	543.41
	2	437.80	431.35	417.22	426.11
	3	414.04	417.31	527.44	455.55
	4	448.59	410.89	578.99	446.96
6	1	612.64	714.53	509.08	541.95
	2	561.52	636.35	425.95	431.91
	3	556.79	632.74	541.87	463.53
	4	516.50	597.11	559.70	453.81

จากตารางที่ 9 เป็นผลการวัดประสิทธิภาพของระบบงาน File Server ที่ได้จากโปรแกรม FFSB ซึ่งโปรแกรมนี้ให้ผลลัพธ์เป็นประสิทธิภาพของเครื่องแม่ข่ายจำนวน 4 เครื่องต่อการทดสอบ 1 ครั้ง ซึ่งภายหลังจากการทดสอบเป็นจำนวน 6 ครั้ง พบว่าเมื่อ Enable Hardware Assisted Virtualization จะมีประสิทธิภาพการทำงานของระบบลดลงทั้งใน VMWare และ MS Hyper-V

ตารางที่ 10: ตารางบันทึกผลผลการทดลองระบบงาน Idle Server

ครั้งที่	เครื่องที่	VMWare		Hyper-V	
		Disable	Enable	Disable	Enable
1	1	17.90	19.20	22.70	20.20
	2	25.60	19.60	19.60	20.60
	3	25.40	27.30	26.50	28.50
	4	25.30	28.10	24.10	23.80

(ตารางมีต่อ)

ตารางที่ 10 (ต่อ): ตารางบันทึกผลผลการทดลองระบบงาน Idle Server

ครั้งที่	เครื่องที่	VMWare		Hyper-V	
		Disable	Enable	Disable	Enable
2	1	19.30	17.10	25.20	19.50
	2	25.90	25.10	18.00	21.70
	3	25.30	25.20	25.90	31.80
	4	23.70	26.80	23.60	20.40
3	1	20.30	17.10	22.50	20.30
	2	24.80	25.10	22.40	27.40
	3	24.90	25.20	23.70	21.40
	4	24.10	26.80	24.30	24.20
4	1	20.00	19.10	27.00	23.10
	2	24.30	24.70	17.90	28.80
	3	25.00	25.30	24.20	20.00
	4	24.80	25.10	23.70	21.40
5	1	21.50	19.00	20.90	24.00
	2	24.20	25.60	24.00	21.10
	3	23.60	24.60	24.30	22.30
	4	24.70	24.90	23.70	26.00
6	1	19.70	19.50	27.80	21.40
	2	24.20	24.60	17.00	23.70
	3	25.20	24.60	24.00	22.10
	4	25.00	25.40	24.30	26.20

จากตารางที่ 10 เป็นผลการวัดประสิทธิภาพของระบบงาน Idle Server ที่ได้จากโปรแกรม Iperf ซึ่งโปรแกรมนี้ให้ผลลัพธ์เป็นประสิทธิภาพของเครื่องแม่ข่ายจำนวน 4 เครื่องต่อการทดสอบ 1 ครั้ง ซึ่งภายหลังจากการทดสอบเป็นจำนวน 6 ครั้ง พบว่าเมื่อ Enable Hardware Assisted Virtualization จะมีประสิทธิภาพการทำงานของระบบเพิ่มขึ้นทั้งใน VMWare และ MS Hyper-V

4.2 ผลการเปรียบเทียบประสิทธิภาพของ Hypervisor เมื่อใช้งานร่วมกับ Hardware Assisted Virtualization

จากการวัดประสิทธิภาพของระบบงานภายใต้สภาพแวดล้อมที่ออกแบบไว้เพื่อเปรียบเทียบการทำงานของ VMware และ MS Hyper-V แล้วนำมาคำนวณเปรียบเทียบอัตราส่วนของประสิทธิภาพที่เปลี่ยนแปลงไป พบว่าเมื่อใช้งานร่วมกับ Hardware Assisted Virtualization กลับทำให้ประสิทธิภาพในการทำงานของระบบงานส่วนใหญ่มีค่าต่ำลง มีเพียงระบบงานที่เป็น Mail Server เท่านั้นที่มีประสิทธิภาพสูงขึ้น โดยใน VMware จะมีประสิทธิภาพเพิ่มขึ้น 8.25% และ MS Hyper-V มีประสิทธิภาพเพิ่มขึ้น 6.17% ในส่วนการวัดประสิทธิภาพการรับ-ส่งข้อมูลบนเครือข่ายแม้จะมีประสิทธิภาพสูงขึ้นแต่พบว่ามีผลต่างเพียงเล็กน้อยเท่านั้นคือ 0.06% ใน VMware และ 0.52% ใน MS Hyper-V สำหรับระบบงานที่มีค่าประสิทธิภาพลดลงมากที่สุดคือ ระบบ Database Server โดยพบว่าใน VMware มีประสิทธิภาพลดลง 5.80% และ MS Hyper-V มีประสิทธิภาพลดลง 9.40% ดังรายละเอียดในตารางที่ 11 และ ตารางที่ 12

ผลการวัดประสิทธิภาพที่ได้สอดคล้องกับสมมติฐานในข้อที่ 1 คือ Hypervisor จากสองค่ายมีอัตราการลดลงหรือเพิ่มขึ้นของประสิทธิภาพในระดับที่ต่างกันเมื่อใช้งานร่วมกับ Hardware Assisted Virtualization และยังสอดคล้องกับสมมติฐานในข้อที่ 2 คือการใช้เทคนิคนี้ ไม่ทำให้ Hypervisor ทำงานได้มีประสิทธิภาพมากขึ้นในทุกสภาพแวดล้อมของระบบงานด้านไอที

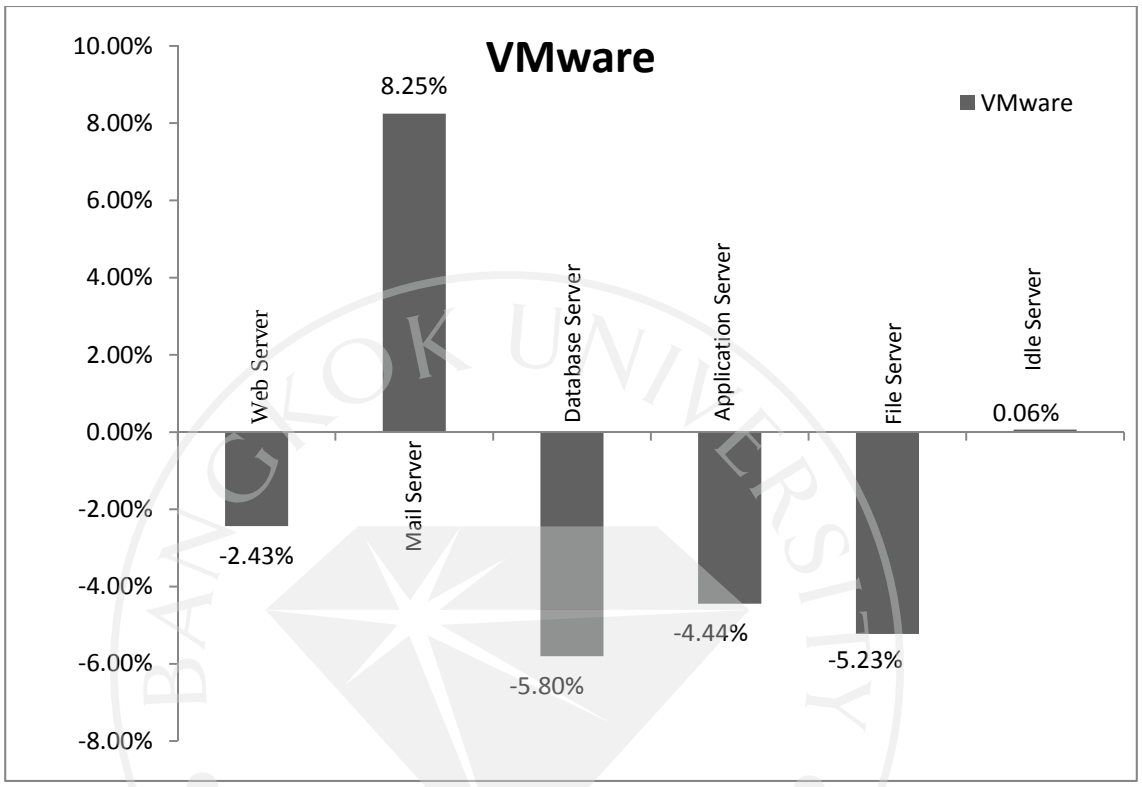
ตารางที่ 11: ตารางเปรียบเทียบผลการวัดประสิทธิภาพแยกตามระบบงานของ VMWare

ระบบงาน	Disable Hardware Assisted	Enable Hardware Assisted	ผลต่างเทียบเป็นร้อยละ
Web Server	21.3600	20.8400	-2.43%
Mail Server	337.5600	365.4000	8.25%
Database Server	76.2835	71.8564	-5.80%
Application Server	28.8000	27.5200	-4.44%
File Server	522.7080	495.3915	-5.23%
Idel Server	23.5250	23.5400	0.06%

ตารางที่ 12: ตารางเปรียบเทียบผลการวัดประสิทธิภาพแยกตามระบบงานของ MS Hyper-V

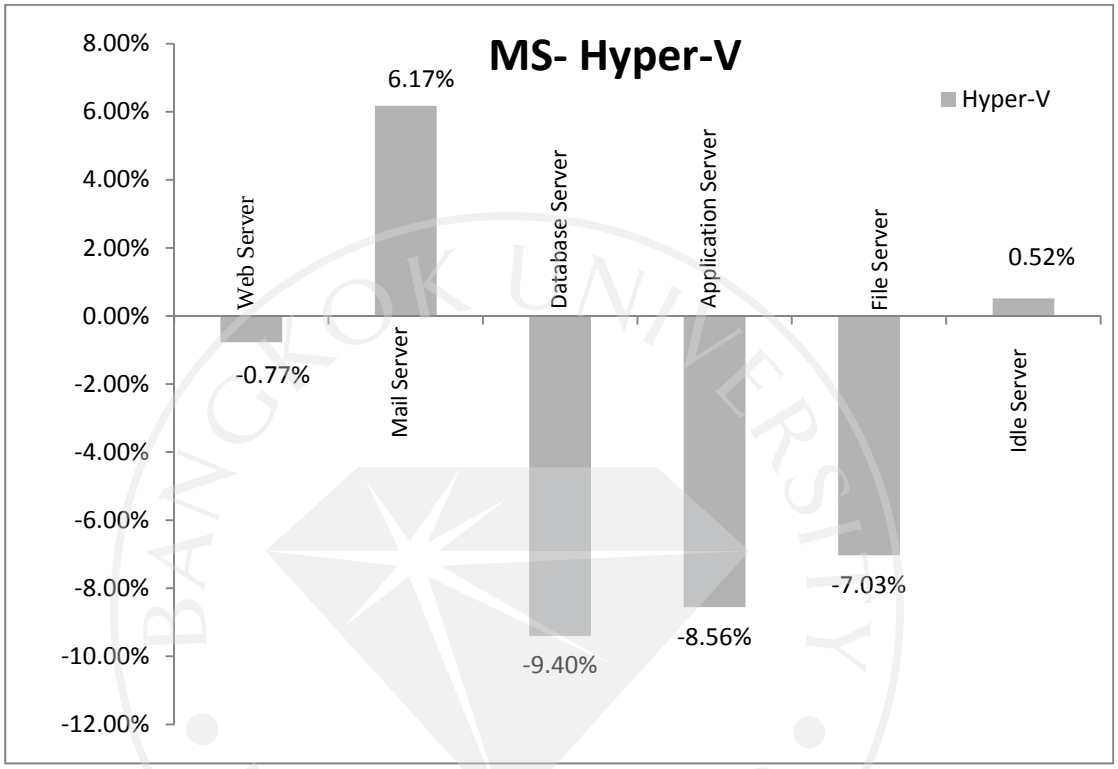
ระบบงาน	Disable Hardware Assisted	Enable Hardware Assisted	ผลต่างเทียบเป็นร้อยละ
Web Server	20.7600	20.6000	-0.77%
Mail Server	373.2000	396.2400	6.17%
Database Server	13.2782	12.0296	-9.40%
Application Server	3.7400	3.4200	-8.56%
File Server	505.3495	469.8175	-7.03%
Idel Server	23.2200	23.3400	0.52%

ภาพที่ 11: แผนภูมิเปรียบเทียบอัตราการเปลี่ยนแปลงของประสิทธิภาพในแต่ละระบบงาน เมื่อใช้เทคนิค Hardware Assisted Virtualization ของ VMware



จากภาพที่ 11 จะเห็นว่า VMWare เมื่อใช้เทคนิค Hardware Assisted Virtualization แต่ละระบบงานจะมีอัตราการเปลี่ยนแปลงของประสิทธิภาพการทำงานต่างกัน โดยมีอัตราการลดลงของประสิทธิภาพใน 4 ระบบงานคือ Web Server, Database Server, Application Server, File Server และมีอัตราการเพิ่มขึ้นของประสิทธิภาพใน 2 ระบบงานคือ Mail Server และ Idle Server

ภาพที่ 12: แผนภูมิเปรียบเทียบอัตราการเปลี่ยนแปลงของประสิทธิภาพในแต่ละระบบงาน
เมื่อใช้เทคนิค Hardware Assisted Virtualization ของ MS Hyper-V



จากภาพที่ 12 จะเห็นว่า MS Hyper-V เมื่อใช้เทคนิค Hardware Assisted Virtualization แต่ละระบบงานจะมีอัตราการเปลี่ยนแปลงของประสิทธิภาพการทำงานต่างกัน โดยมีอัตราการลดลงของประสิทธิภาพใน 4 ระบบงานคือ Web Server, Database Server, Application Server, File Server และมีอัตราการเพิ่มขึ้นของประสิทธิภาพสูงกว่าใน 2 ระบบงานคือ Mail Server และ Idle Server

บทที่ 5
บทสรุปและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงบทสรุปภาพรวมของการศึกษาวิจัยและการอภิปรายผลตามหัวข้อที่ได้ดำเนินการศึกษาวิจัย รวมถึงนำเสนอข้อเสนอแนะทั้งในด้านการนำไปใช้ประโยชน์ และข้อเสนอแนะในการศึกษาค้นคว้าเพิ่มเติม โดยมีรายละเอียดดังต่อไปนี้

5.1. วัตถุประสงค์

5.1.1. สามารถนำผลการวิจัยมาเป็นข้อพิจารณาในการเลือกใช้เทคโนโลยี ทั้งในส่วนของฮาร์ดแวร์และซอฟต์แวร์ได้อย่างเหมาะสมกับองค์กร

5.1.2. สามารถนำมาใช้เป็นแนวทางในการวางโครงสร้างพื้นฐานระบบไอทีแบบ Virtualization ได้อย่างเหมาะสมกับความต้องการขององค์กร

5.1.3. ทำให้ทราบได้ว่าประสิทธิภาพของระบบงานด้านไอทีแต่ละประเภทเมื่อใช้เทคนิค Hardware Assisted Virtualization จะมีอัตราการเปลี่ยนแปลงของประสิทธิภาพแตกต่างกัน

5.2. สมมติฐานการวิจัย

ผลการศึกษาวิจัยสามารถตอบสมมติฐานของงานวิจัยได้ดังนี้

5.2.1. Hypervisor สองชนิดคือ VMware และ MS Hyper-V เมื่อใช้เทคนิค Hardware Assisted Virtualization จะมีอัตราการเปลี่ยนแปลงของประสิทธิภาพการทำงานต่างกัน

5.2.2. เทคนิคของ Hardware Assisted Virtualization ไม่ทำให้ Hypervisor มีประสิทธิภาพการทำงานสูงขึ้นในทุกประเภทของระบบงานด้านไอที

5.3. กระบวนการศึกษา

ผู้ศึกษาวิจัยได้ดำเนินการสร้างสภาพแวดล้อม เพื่อจำลองระบบเครื่องแม่ข่ายและ Workload ของระบบงานทางด้านไอทีที่ทำงานแบบ Virtualization ขึ้นมาทดสอบ โดยออกแบบการทดสอบให้มีสภาพแวดล้อมของ อุปกรณ์ฮาร์ดแวร์ ระบบปฏิบัติการที่เป็น Guest OS และซอฟต์แวร์ที่ให้บริการระบบงานเหมือนกัน แตกต่างกันเพียงใช้ Hypervisor ต่างชนิดกันคือ VMware และ MS Hyper-V จากนั้นจะวัดประสิทธิภาพการทำงานของ Hypervisor แต่ละชนิด ในสองลักษณะคือเมื่อ

Disable และ Enable Hardware Assisted Virtualization ใน CPU โดยการแก้ไขค่า Configuration ในหัวข้อ Intel VT-d Technology และ I/O Virtualization ที่ BIOS ของเครื่องแม่ข่าย แล้วนำผลที่ได้จากการวัดประสิทธิภาพมาเปรียบเทียบหาอัตราการเพิ่มขึ้นหรือลดลงของประสิทธิภาพในการทำงานในแต่ละระบบ

5.4. อภิปรายผลการศึกษาค้นคว้า

จากผลการศึกษาวิจัยพบว่า เทคนิคการสร้างเครื่องแม่ข่ายเสมือนบนสถาปัตยกรรม CPU x86 ในแบบ Hardware Assisted Virtualization ณ ปัจจุบันยังไม่ได้ช่วยเพิ่มประสิทธิภาพของเครื่องแม่ข่ายเสมือน แม้ว่าจากการศึกษาเทคนิคการสร้างเครื่องแม่ข่ายเสมือนจะพบว่าเทคนิคนี้เอื้อประโยชน์ต่อผู้ผลิตซอฟต์แวร์ Hypervisor ด้วยการเพิ่มชุดคำสั่งพิเศษใน CPU ให้สามารถบริหารจัดการ Virtual Machine ได้สะดวกขึ้น

จึงเป็นข้อพิจารณาให้กับผู้ที่ต้องการเลือกใช้ซอฟต์แวร์ประเภท Hypervisor ว่าประสิทธิภาพของระบบเครื่องแม่ข่ายเสมือนไม่สามารถอาศัยเพียงแต่การพัฒนา CPU ให้สนับสนุนการทำ Virtualization ได้เพียงอย่างเดียว จึงควรเลือกใช้งานซอฟต์แวร์จากผู้ผลิตซอฟต์แวร์ที่มีศักยภาพการในพัฒนาซอฟต์แวร์ของตนเองอย่างต่อเนื่องเพราะจะเห็นได้ว่าเทคโนโลยีนี้จำเป็นต้องพัฒนาอยู่ตลอดเวลา นอกจากนี้เมื่อย้อนกลับไปพิจารณาวัตถุประสงค์ของการออกแบบเทคนิค Hardware Assisted Virtualization จะพบว่า การพัฒนาในปัจจุบันมุ่งที่จะพัฒนาให้ CPU แบบ x86 สามารถสร้างระบบ Virtualization ได้เช่นเดียวกับ IBM S/370 เพื่อประโยชน์ในด้านการบริหารจัดการ Virtual Machine และเสถียรภาพของระบบ ไม่ได้เน้นไปในด้านของการเพิ่มประสิทธิภาพโดยเฉพาะ จึงคาดว่าน่าจะเป็นสาเหตุที่ทำประสิทธิภาพในบางระบบงานลดลง

5.5. ข้อเสนอแนะเพื่อการนำไปใช้

ผลการวิจัยครั้งนี้สามารถเป็นประโยชน์ต่อผู้ให้บริการศูนย์ข้อมูล ที่มีเครื่องแม่ข่ายเสมือนให้บริการอยู่คือ ในเครื่องแม่ข่ายที่ให้บริการระบบงานประเภท Web Server, Database Server, Application Server และ File Server การไม่เปิดใช้งาน Hardware Assisted Virtualization ระบบเครื่องแม่ข่ายเสมือนจะทำงานได้มีประสิทธิภาพมากกว่า แต่สำหรับในระบบงานที่เป็น Mail Server การเปิดใช้งานเทคนิคนี้จะทำให้ระบบเครื่องแม่ข่ายเสมือนมีประสิทธิภาพมากกว่า

5.6. ข้อเสนอแนะเพื่อการศึกษาวิจัยในอนาคต

ในการศึกษาวิจัยครั้งนี้ พบปัญหาว่าเมื่อวัดประสิทธิภาพของ Hypervisor แต่ละประเภทออกมาในสองลักษณะคือ Disable และ Enable Hardware Assisted Virtualization แล้วการที่จะนำเอาอัตราการเปลี่ยนแปลงของประสิทธิภาพการทำงานของ Hypervisor มาเปรียบเทียบกันโดยตรง อาจจะทำให้สร้างความสับสนในแง่ของตัวเลข เนื่องจากการคำนวณอัตราการเปลี่ยนแปลง

ของประสิทธิภาพเทียบเป็นเปอร์เซ็นต์ จะนำค่าที่วัดในสภาวะ Disable Hardware Assisted Virtualization มาเป็นค่าตั้งต้นในการเปรียบเทียบ ดังนั้นหากระบบงานใดมีค่าตั้งต้นที่ต่างกันมากๆ จะทำให้ดูเหมือนว่าตัวเลขผลต่างของประสิทธิภาพที่วัดได้จริง อาจจะไม่สอดคล้องกับอัตราการเปลี่ยนแปลงเมื่อคิดเป็นเปอร์เซ็นต์

ดังนั้นจึงคิดว่าแนวทางในการศึกษาเพิ่มเติมในอนาคต การเปรียบเทียบอัตราการเปลี่ยนแปลงของประสิทธิภาพอาจจะทำในลักษณะที่ใช้เครื่องแม่ข่ายที่ไม่ได้ทำ Virtualization มาเป็นฐานในการเปรียบเทียบโดย สร้างสภาพแวดล้อมของการวัดประสิทธิภาพเป็น 3 สภาพแวดล้อม คือ 1. เครื่องแม่ข่ายจริง 2. เครื่องแม่ข่ายที่ใช้งาน VMWare 3. เครื่องแม่ข่ายที่ใช้งาน MS Hyper-V จากนั้นเมื่อได้ค่าของการวัดประสิทธิภาพออกมาให้ใช้ค่าประสิทธิภาพที่วัดได้จากเครื่องแม่ข่ายจริงมาเป็นฐานในการเปรียบเทียบกับ Hypervisor ในแต่ละชนิด ซึ่งจะทำให้เห็นความชัดเจนในด้านตัวเลขอัตราการเปลี่ยนแปลงของประสิทธิภาพมากยิ่งขึ้น

บรรณานุกรม

Journal/Articles

Bhukya, D. P., & Ramachandram, S. (2009). Performance evaluation of virtualization and non virtualization on different workloads using doe methodology. IACSIT International Journal of Engineering and Technology, 1(5), 404-407.

Bichler, M., Setzer, T., & Speitkamp, B. (2006). Capacity planning for virtualized servers. Workshop on Information Technologies and Systems (WITS). Wisconsin, USA.

Chen, Q., & Xin, R. (2005). Optimizing enterprise IT infrastructure through virtual server consolidation. The 2005 Informing Science and IT Education Joint Conference. New Jersey: Informing Science.

Chen, W., Hongyi, L., Li, S., Zhiying, W., Nong, X., & Dan, C. (2003). A novel hardware assisted full virtualization technique. The 9th International Conference for Young Computer Scientists, 18-21 November 2008, Hunan, 1292-1297.

Farr, E. M., Harper, R. E., Spainhower, L. F., & Xenidis, J. (2008). A case for high availability in a virtualized environment (HAVEN). The Third International Conference on Availability, Reliability and Security, 4-7 March 2008, Barcelona, Spain, 675 - 682

Fisher-Ogden, J. (2006). Hardware support for efficient virtualization. San Diego. University of California.

Lu, J., Makhlis, L., & Chen, J. (2006). Measuring and modeling the performance of the xen VMM. 32th International Conference Computer Measurement Group. 3-6 December 2006, Reno, Nevada, USA.

Makhija, V., Herndon, B., Smith, P., Roderick, L., Zamost, E., & Anderson, J. (2006). VMmark: A scalable benchmark for virtualized systems. Technical Report VMware-TR-2006-002. Palo Alto, CA: VMware.

Neiger, G., Santoni, A., Leung, F., Rodgers, D., & Uhlig, R. (2006, August). Intel® virtualization technology: Hardware support for efficient processor virtualization. Intel Technology Journal . Intel Press.

Padala, P., Zhu, X., Wang, Z., Singhal, S., & Shin, K. G. (2007, April 11). Performance evaluation of virtualization technologies for server. Hewlett-Packard Development Company.

Rose, R. (2004, March 8). Survey of system virtualization techniques. Oregon State University.

Sundarrajan, S., & Nellitheertha, H. (2006, December). Server consolidation and virtualization. Infosys.

Umeno, H., Paraynot, M. L., Teramoto, K., Kawanot, M., & Inamasu, H. (2006). Performance evaluation on server consolidation using virtual machines. SICE-ICASE International Joint Conference 2006. 18-21 October 2006, Busan, Korea, 2730 - 2734.

Theurer, A., Rister, K., & Dobbelstein, S. (2008). A survey of virtualization workloads. Proceedings of the Linux Symposium Vol. 2, 215.

VMware Inc. (2007). Understanding full virtualization, paravirtualization, and hardware assist. Palo Alto, CA: VMware.

Wolf, C. (2007). Let's get virtual: A look at today's server virtualization architectures. Utah, USA: Burton Group.

Theses Dissertations and Papers

Marinescu, V. V. (2008, February 29). Design and evaluation of self-management approaches for virtual machine-based environments. Unpublished master's thesis,Wiesbaden University of Applied Sciences,Wiesbaden,Germany.

Internet

Adams, K., & Agesen, O. (2006). A comparison of software and hardware techniques for x86 virtualization. Retrieved by 10 November 2010 from http://www.vmware.com/pdf/asplos235_adams.pdf

Apache Software Foundation. (n.d.). JMeter. Retrieved by 10 November 2010 from http://jakarta.apache.org/jmeter/

Chris, M. (2010, January 19). Virtualization and cloud will transform IT. Retrieved by 8 October 2010 from http://www.informationweek.com/blog/main/archives/2010/01/gartner_virtual.html

Flexible file system benchmark. (n.d.). Retrieved by 20 November 2010 from http://sourceforge.net/projects/ffsb

Georges, A., & Eeckhout, L. (2010). Performance metrics for consolidate servers. Retrieved by 20 November 2010 from http://itkovian.net/base/files/papers/hpcvirt-2010-ageorges-preprint.pdf

HEISSLER, A. (2010, September). Performance analysis of xen virtual machines in real-world scenarios. Retrieved by 20 November 2010 from http://arxiv.org/abs/1009.5878

Iperf. (n.d.). Retrieved by 22 November 2010 from http://sourceforge.net/projects/iperf

Johan, D. G. (2008). Hardware accelerated virtualization: Intel VT-x and AMD. Retrieved by 8 March 2010 from http://www.anandtech.com/show/2480/9

Rajesh, D. (2010). Hyper-V VM processor compatibility in brief. Retrieved by 2 October 2010 from http://download.microsoft.com/download/F/2/1/F2146213-4AC0-4C50-B69A-12428FF0B077/VM%20processor%20compatibility%20mode.doc

Standard Performance Evaluation Corporation. (2010, June). SPECvirt_sc2010 design document. Retrieved by 2 September 2010 from http://www.spec.org/virt_sc2010/docs/SPECvirt_Design_Overview.html

The PostgreSQL Global Development Group. (n.d.). PostgreSQL 9.1devel documentation. Retrieved by 12 November 2010 from: http://developer.postgresql.org/pgdocs/postgres/pgbench.html

VMware Inc. (2009, December 21). VMware products and hardware-assisted virtualization (VT-x/AMD-V). Retrieved by 2 October 2010 from http://communities.vmware.com/docs/DOC-9150

ประวัติผู้เขียน

ชื่อ-นามสกุล นายณัฐกร เฉยศิริ

อีเมล์ nattakorn.c@bu.ac.th

ประวัติการศึกษา วท.บ. (วิทยาศาสตร์)
มหาวิทยาลัยบูรพา

ประสบการณ์ทำงาน

2541-2542 นักเขียนโปรแกรม บริษัทประชาอาภรณ์ จำกัด มหาชน

2542-ปัจจุบัน อาจารย์ประจำมหาวิทยาลัยกรุงเทพ

BANGKOK UNIVERSITY FOUNDED 1962

มหาวิทยาลัยกรุงเทพ

ข้อตกลงว่าด้วยการอนุญาตให้ใช้สิทธิในวิทยานิพนธ์/สารนิพนธ์

วันที่ 25 เดือน กรกฎาคม พ.ศ. 2554

ข้าพเจ้า (นาย/~~นาง/นางสาว~~) ณัฐกร เฉยศิริ อยู่บ้านเลขที่ 509/79 ม.9
ซอย — ถนน — ตำบล/แขวง หนองปรือ
อำเภอ/เขต บางละมุง จังหวัด ชลบุรี รหัสไปรษณีย์ 20260
เป็นนักศึกษาของมหาวิทยาลัยกรุงเทพ รหัสประจำตัว 7520700159
ระดับปริญญา ☐ ตรี ☑ โท ☐ เอก
หลักสูตร วิทยาศาสตรมหาบัณฑิต สาขาวิชา เทคโนโลยีสารสนเทศและการจัดการ
คณะ บัณฑิตวิทยาลัย ซึ่งต่อไปนี้เรียกว่า **"ผู้อนุญาตให้ใช้สิทธิ"** ฝ่ายหนึ่ง และ

มหาวิทยาลัยกรุงเทพ ตั้งอยู่เลขที่ 119 ถนนพระราม 4 แขวงพระโขนง เขตคลองเตย กรุงเทพมหานคร 10110 ซึ่งต่อไปนี้เรียกว่า **"ผู้ได้รับอนุญาตให้ใช้สิทธิ"** อีกฝ่ายหนึ่ง

ผู้อนุญาตให้ใช้สิทธิ และ ผู้ได้รับอนุญาตให้ใช้สิทธิ ตกลงทำสัญญากันโดยมีข้อความดังต่อไปนี้

ข้อ 1. ผู้อนุญาตให้ใช้สิทธิขอรับรองว่าเป็นผู้สร้างสรรค์และเป็นผู้มีสิทธิแต่เพียงผู้เดียวในงานสารนิพนธ์/วิทยานิพนธ์หัวข้อ การวัดและเปรียบเทียบประสิทธิภาพ Hypervisor ที่ทำงานร่วมกับ Hardware Assisted Virtualization
ซึ่งถือเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิทยาศาสตรมหาบัณฑิต ของมหาวิทยาลัยกรุงเทพ (ต่อไปนี้เรียกว่า **"สารนิพนธ์/วิทยานิพนธ์"**)

ข้อ 2. ผู้อนุญาตให้ใช้สิทธิตกลงยินยอมให้ผู้ได้รับอนุญาตให้ใช้สิทธิโดยปราศจากค่าตอบแทนและไม่มีกำหนดระยะเวลาในการนำสารนิพนธ์/วิทยานิพนธ์ ซึ่งรวมถึงแต่ไม่จำกัดเพียงการทำซ้ำ ดัดแปลง เผยแพร่ต่อสาธารณชน ให้เช่าต้นฉบับหรือสำเนางาน ให้ประโยชน์อันเกิดจากลิขสิทธิ์แก่ผู้อื่น อนุญาตให้ผู้อื่นใช้สิทธิโดยจะกำหนดเงื่อนไขอย่างหนึ่งอย่างใดด้วยหรือไม่ก็ได้ ไม่ว่าทั้งหมดหรือเพียงบางส่วน หรือการกระทำอื่นใดในลักษณะทำนองเดียวกัน

ข้อ 3. หากกรณีมีข้อขัดแย้งในปัญหาลิขสิทธิ์ในสารนิพนธ์/วิทยานิพนธ์ระหว่างผู้อนุญาตให้ใช้สิทธิกับบุคคลภายนอกก็ดี หรือระหว่างผู้ได้รับอนุญาตให้ใช้สิทธิกับบุคคลภายนอกก็ดี หรือมีเหตุขัดข้องอื่นๆ เกี่ยวกับลิขสิทธิ์ อันเป็นเหตุให้ผู้รับอนุญาตให้ใช้สิทธิไม่สามารถนำงานนั้นออกทำซ้ำ เผยแพร่ หรือโฆษณาได้ ผู้อนุญาตให้ใช้สิทธิยินยอมรับผิดและชดใช้ค่าเสียหายแก่ผู้ได้รับอนุญาตให้ใช้สิทธิในความเสียหายต่างๆ ที่เกิดขึ้นแก่ผู้ได้รับอนุญาตให้ใช้สิทธิทั้งสิ้น

สัญญานี้ทำขึ้นสองฉบับ มีข้อความเป็นอย่างเดียวกัน คู่สัญญาได้อ่านและเข้าใจข้อความในสัญญานี้โดยละเอียดแล้ว จึงได้ลงลายมือชื่อให้ไว้เป็นสำคัญต่อหน้าพยาน และเก็บรักษาไว้ฝ่ายละฉบับ

ลงชื่อ.....ผู้อนุญาตให้ใช้สิทธิ

(นายณัฐกร เฉยศิริ)

ลงชื่อ.....ผู้ได้รับอนุญาตให้ใช้สิทธิ

( )

ผู้อำนวยการสำนักหอสมุด

ลงชื่อ.....พยาน

(ผู้ช่วยศาสตราจารย์ ดร. ศิวพร หวังพิพัฒน์วงศ์)

คณบดีบัณฑิตวิทยาลัย

ลงชื่อ.....พยาน

( )