การพัฒนาระบบจัดเก็บข้อมูลบนเครือข่ายสำหรับเครือข่ายขนาดเล็กด้วยเทคโนโลยีสำเร็จรูป

นายวีรภัทร พรหมชนะ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2554

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ที่ส่งผ่านทางบัณฑิตวิทยาลัย
The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository(CUIR)
are the thesis authors' files submitted through the Graduate School.

DEVELOPMENT OF NETWORK STORAGE FOR SMALL NETWORK USING OFF-THE-SHELF TECHNOLOGY

Mr. Veerapat Phromchana

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Engineering Program in Computer Engineering

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2011

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การพัฒนาระบบจัดเก็บข้อมูลบนเครือข่ายสำหรับเครือข่ายขนาดเล็กด้วยเทคโนโลยีสำเร็จรูป
โดย	นายวีรภัทร พรหมชนะ
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร.ณัฐวุฒิ หนูไพโรจน์
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม	ผู้ช่วยศาสตราจารย์ ดร.เกริก ภิรมย์โสภา

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(อาจารย์ ดร.ยรรยง เต็งอำนวย)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ ดร.ณัฐวุฒิ หนูไพโรจน์)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม
(ผู้ช่วยศาสตราจารย์ ดร.เกริก ภิรมย์โสภา)

..... กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร.ภุชงค์ อุทโยภาศ)

วีรภัทร พรหมชนะ : การพัฒนาระบบจัดเก็บข้อมูลบนเครือข่ายสำหรับเครือข่ายขนาดเล็กด้วยเทคโนโลยีสำเร็จรูป. (DEVELOPMENT OF NETWORK STORAGE FOR SMALL NETWORK USING OFF-THE-SHELF TECHNOLOGY) อ.ที่ปรึกษาวิทยานิพนธ์หลัก : ผศ. ดร.ณัฐวุฒิ หนูไพโรจน์, อ.ที่ปรึกษาวิทยานิพนธ์ร่วม : ผศ. ดร.เกริก ภิรมย์โสภา 45 หน้า.

แนวโน้มของข้อมูลต่างๆ ที่มุ่งหน้าสู่ยุคดิจิทัล และอุปกรณ์ที่สามารถเชื่อมต่อระบบเครือข่ายได้ค่อยๆ กลายมาเป็นเรื่องปกติในปัจจุบัน ส่งผลให้เกิดความต้องการในการจัดเก็บและแบ่งปันข้อมูลขึ้นอย่างมากรวมถึงผู้ใช้ตามบ้าน หรือสำนักงานขนาดเล็ก เครื่องบริการไฟล์มีบทบาทสำคัญในการเป็นตัวกลางในการจัดเก็บข้อมูล แต่เมื่อเวลาผ่านไปข้อมูลได้เพิ่มขึ้นเรื่อยๆ ทำให้พื้นที่จัดเก็บข้อมูลไม่เพียงพอต่อความต้องการ การขยายพื้นที่จัดเก็บข้อมูลนั้นทำได้ลำบากหรือยุ่งยากต่อการบริหารจัดการ ส่วนระบบจัดเก็บข้อมูลสำหรับภาคธุรกิจขนาดใหญ่นั้นมีต้นทุนที่สูงเกินไปสำหรับเครือข่ายขนาดเล็ก การนำเทคโนโลยีสำเร็จรูปที่มีอยู่มากมายอย่างเช่น อุปกรณ์สำหรับผู้ใช้ตามบ้าน หรือซอฟต์แวร์โอเพนซอร์ส มาประยุกต์ใช้ให้เกิดประสิทธิภาพสูงสุดจะช่วยลดข้อจำกัดต่างๆ เหล่านี้ได้

งานวิจัยนี้นำเสนอแบบจำลองของ ระบบจัดเก็บข้อมูลบนเครือข่ายแบบขยายได้ ซึ่งคือระบบบันทึกข้อมูลบนเครือข่ายที่สามารถเพิ่มอุปกรณ์บันทึกข้อมูลได้โดยอัตโนมัติ และให้ประสิทธิภาพสูงสุดในการใช้งานต่างๆ โดยงานวิจัยนี้ได้วิเคราะห์ส่วนประกอบต่างๆที่มีอยู่แพร่หลายและเหมาะในการสร้างระบบบันทึกบนเครือข่ายแบบขยายได้ ซึ่งได้แก่ ระบบไฟล์ โพรโตคอลในการเชื่อมต่อฮาร์ดดิสก์ ความเร็วของระบบเครือข่าย และรูปแบบการเชื่อมต่อเครือข่าย

ภาควิชา .....วิศวกรรมคอมพิวเตอร์..... ลายมือชื่อนิสิต .....

สาขาวิชา.....วิศวกรรมคอมพิวเตอร์..... ลายมือชื่อ อ. ที่ปรึกษาวิทยานิพนธ์หลัก.....

ปีการศึกษา.....2554..... ลายมือชื่อ อ. ที่ปรึกษาวิทยานิพนธ์ร่วม.....

5070707021 : MAJOR COMPUTER ENGINEERING
KEYWORD : STORAGE / NETWORK STORAGE / ZEROCONF

VEERAPAT PHROMCHANA : DEVELOPMENT OF NETWORK STORAGE FOR SMALL NETWORK USING OFF-THE-SHELF TECHNOLOGY. ADVISOR : ASST. PROF. NATAWUT NUPAIROJ, Ph.D., CO-ADVISOR : ASST. PROF. KRERK PIROMSOPA, Ph.D., 45 pp.

The emerging trends of digital-data and network equipped devices have become common. This leads to the needs for storing and sharing contents in large-scale even home or small office. A file server plays very important role in content sharing. However, the data are increasing over time in limited storage spaces. Increasing storage spaces are also difficult to manage. And enterprise class storage system cost too high for small network. Utilizing of Off-the-shelf technology that are available widely such as home use equipment and open source software can lower these limits.

This research proposes a model of Extensible Network Storage in which a network storage system that can automatically detect a newly added storage device and merge in to an existing system and also provide maximum efficiency. In this research, analyzing components that are widely available and suitable for extensible network storage such as file system, protocol, network speed and network topology.

Department: Computer Engineering Student's Signature .....
Field of Study: Computer Engineering Advisor's Signature .....
Academic Year: 2011 Co-advisor's Signature .....

กิตติกรรมประกาศ

ขอขอบพระคุณผู้ช่วยศาสตราจารย์ ดร.ณัฐวุฒิ หนูไพโรจน์ และผู้ช่วยศาสตราจารย์ ดร.เกริก ภิรมย์โสภา ที่ให้คำปรึกษา คำแนะนำ ช่วยจุดประกายความคิด และความช่วยเหลือต่าง ๆ มาโดยตลอดระยะเวลาการศึกษาและการวิจัย รวมทั้งสนับสนุนอุปกรณ์ต่างๆ ในการดำเนินการวิจัย ทำให้งานวิจัยนี้สำเร็จลุล่วงได้ด้วยดี

ขอขอบพระคุณอาจารย์ ดร.ยรรยง เต็งอำนวย และผู้ช่วยศาสตราจารย์ ดร.ภุชงค์ อุทโยภาศ คณะกรรมการสอบวิทยานิพนธ์ที่ให้คำชี้แนะในการปรับปรุงงานวิทยานิพนธ์ในด้านต่างๆ

ขอกราบขอบพระคุณคุณพ่อ คุณแม่ ที่ให้ความรัก ความห่วงใย ทำให้มีความสุขทั้งกายและใจ และเป็นกำลังใจในการดำเนินชีวิตมาโดยตลอด

ขอขอบคุณคุณสุดาสิรี ขอบคุณเพื่อนๆ พี่ๆ น้องๆ ทุกคนที่เป็นกำลังใจให้ตลอดมา เพื่อนๆ พี่ๆ น้องๆ ที่ห้องปฏิบัติการวิศวกรรมระบบสารสนเทศ ภาควิชาวิศวกรรมคอมพิวเตอร์ทุกคน ที่ร่วมทุกข์ร่วมสุข แบ่งปันความรู้ แนวคิดต่าง และบรรยากาศที่ดี ตลอดระยะเวลาที่ดำเนินการวิจัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ญ

สารบัญภาพ ..... ฏ

บทที่

1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ของการวิจัย 2

1.3 ขอบเขตของการวิจัย 2

1.4 ประโยชน์ที่คาดว่าจะได้รับ 3

1.5 วิธีดำเนินการวิจัย 3

1.6 ผลงานตีพิมพ์ 3

2 เอกสารและงานวิจัยที่เกี่ยวข้อง 5

2.1 แนวคิดและทฤษฎี 5

2.1.1 การค้นหาอุปกรณ์ 5

2.1.2 การเชื่อมต่อฮาร์ดดิสก์ผ่านระบบเครือข่าย 5

2.1.3 การจัดการพื้นที่เก็บข้อมูล (Volume Manager) 6

2.1.4 ระบบไฟล์ที่ปรับเปลี่ยนขนาดได้ 7

2.2 เอกสารและงานวิจัยที่เกี่ยวข้อง 7

บทที่ หน้า

2.2.1 งานวิจัยที่เกี่ยวข้องทางด้านระบบจัดเก็บข้อมูลที่ขยายพื้นที่จัดเก็บได้ 7
2.2.2 งานวิจัยที่เกี่ยวข้องทางด้านการวัดประสิทธิภาพโพรโทคอลในการรับ-ส่งข้อมูล 8
3 การออกแบบ 9
3.1 แบบจำลองระบบจัดเก็บข้อมูลแบบขยายพื้นที่จัดเก็บข้อมูลโดยอัตโนมัติ 9
3.1.1 การตรวจหาอุปกรณ์ใหม่ 10
3.1.2 การเชื่อมต่อกับฮาร์ดดิสก์ 10
3.1.3 การจัดการพื้นที่สำหรับผู้ใช้ 10
3.1.4 ขั้นตอนการทำงานของเครื่องจัดการ 10
3.2 การวัดประสิทธิภาพส่วนประกอบของระบบ 11
3.2.1 การทดลอง 11
3.2.1 ปัจจัยภายนอก 13
3.2.2 อุปกรณ์ที่ใช้ในการทดสอบ 14
3.2.3 การทดสอบการทำงานของแบบจำลอง 15
4 การทดสอบ 16
4.1 การวัดประสิทธิภาพส่วนประกอบของระบบ 16
4.1.1 ผลการทดลอง 16
4.1.2 ผลกระทบจากปัจจัยภายนอก 32
4.2 การทดสอบการทำงานของแบบจำลองที่ออกแบบ 37
5 บทสรุป 40
5.1 สรุปผลการวิจัย 40
5.2 ปัญหาและข้อจำกัดที่พบจากการวิจัย 41
5.3 ข้อเสนอแนะ 41

หน้า

รายการอ้างอิง..... 42

ประวัติผู้เขียนวิทยานิพนธ์ 45

สารบัญตาราง

ตารางที่ หน้า

3.1 รายละเอียดของส่วนประกอบต่างๆที่ทำการทดสอบ 12

3.2 รายละเอียดอุปกรณ์ที่ใช้ในการทดสอบ 15

4.1 ผลการวัดประสิทธิภาพการเขียนข้อมูลในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที) 18

4.2 ผลการวัดประสิทธิภาพการอ่านข้อมูลในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที) 18

4.3 ผลการวัดประสิทธิภาพการเขียนข้อมูลในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที) 21

4.4 ผลการวัดประสิทธิภาพการอ่านข้อมูลในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที) 21

4.5 ความเร็วในการเขียน-อ่านข้อมูลบนระบบไฟล์ ext3 ที่เครื่องจัดการโดยตรงบนเครือข่ายความเร็วระดับ 100 เมกกะบิตต่อวินาที 23

4.6 ผลการวัดประสิทธิภาพการเขียนข้อมูลในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที) 24

4.7 ผลการวัดประสิทธิภาพการอ่านข้อมูลในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที) 24

4.8 ผลการวัดประสิทธิภาพการเขียนข้อมูลในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที) 27

4.9 ผลการวัดประสิทธิภาพการอ่านข้อมูลในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที) 27

4.10 ความเร็วในการเขียน-อ่านข้อมูลบนระบบไฟล์ ext4 ที่เครื่องจัดการโดยตรงบนเครือข่ายความเร็วระดับ 1 กิกะบิตต่อวินาที 30

สารบัญภาพ

ภาพที่ หน้า

3.1 แบบจำลองของระบบที่ออกแบบ 9

3.2 ขั้นตอนการทำงานของระบบ 11

3.3 เครือข่ายแบบรวมส่วนจัดเก็บข้อมูล (LAN-like model) 12

3.4 เครือข่ายแบบแยกส่วนจัดเก็บข้อมูล (SAN-like model) 12

3.5 โพรโทคอลที่ใช้ในการทดสอบส่วนประกอบ 13

3.6 การทดสอบการรบกวนของระบบ 14

4.1 ความเร็วในการเขียนข้อมูลระหว่างระบบไฟล์และโพรโทคอล 16

4.2 ความเร็วในการอ่านข้อมูลระหว่างระบบไฟล์และโพรโทคอล 17

4.3 ความเร็วในการอ่านไฟล์ขนาด 4 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที 19

4.4 ความเร็วในการเขียนไฟล์ขนาด 350 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที 19

4.5 ความเร็วในการอ่านไฟล์ขนาด 350 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที 20

4.6 ความเร็วในการเขียนไฟล์ขนาด 350 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที 22

4.7 ความเร็วในการอ่านไฟล์ขนาด 350 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที 22

4.8 เปรียบเทียบประสิทธิภาพระบบไฟล์ ext3 ระหว่างรูปแบบการเชื่อมต่อต่างๆ 23

4.9 ความเร็วในการเขียนข้อมูลขนาด 4 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที 25

4.10 ความเร็วในการเขียนข้อมูลขนาด 350 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที 25

ภาพที่ หน้า

4.11 ความเร็วในการอ่านข้อมูลขนาด 1 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที 26

4.12 ความเร็วในการอ่านข้อมูลขนาด 350 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที 26

4.13 ความเร็วในการเขียนข้อมูลขนาด 4 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที 28

4.14 ความเร็วในการเขียนข้อมูลขนาด 350 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที 28

4.15 ความเร็วในการเขียนข้อมูลขนาด 4 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที 29

4.16 ความเร็วในการอ่านข้อมูลขนาด 350 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที 29

4.17 เปรียบเทียบความเร็วในการเขียนข้อมูลบนการเชื่อมต่อเครือข่ายแบบต่างๆ 30

4.18 เปรียบเทียบความเร็วในการอ่านข้อมูลบนการเชื่อมต่อเครือข่ายแบบต่างๆ 30

4.19 เปรียบเทียบความเร็วในการอ่านข้อมูลขนาด 4 เมกกะไบต์ระหว่างการตั้งค่าที่ดีที่สุดของเครือข่ายแต่ละความเร็ว 31

4.20 เปรียบเทียบความเร็วในการอ่านข้อมูลขนาด 350 เมกกะไบต์ระหว่างการตั้งค่าที่ดีที่สุดของเครือข่ายแต่ละความเร็ว 32

4.21 การทดสอบการรบกวนจากเครื่องลูกข่ายที่ทำการทดสอบในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที 33

4.22 การวัดประสิทธิภาพระบบไฟล์บนเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที ด้วยโปรแกรม Intel NASPT 34

4.23 การวัดประสิทธิภาพระบบไฟล์บนเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที โดยมีการรบกวนจากเครื่องลูกข่ายด้วยโปรแกรม Intel NASPT 34

ภาพที่ หน้า

4.24 การวัดประสิทธิภาพระบบไฟล์บนเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที ด้วยโปรแกรม Intel NASPT 35

4.25 การวัดประสิทธิภาพระบบไฟล์บนเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที โดยมีการรบกวนจากเครื่องลูกข่ายด้วยโปรแกรม Intel NASPT 35

4.26 เปรียบเทียบระบบบันทึกข้อมูลที่ไม่มีการรบกวน มีการรบกวนจากเครื่องลูกข่าย และมีการรบกวนจากเครื่องอื่นภายในระบบ ของเครื่องข่ายความเร็วระดับ 1 กิกะบิตต่อวินาที 36

4.27 เปรียบเทียบระบบบันทึกข้อมูลที่ไม่มีการรบกวน มีการรบกวนจากเครื่องลูกข่าย และมีการรบกวนจากเครื่องอื่นภายในระบบ ของเครื่องข่ายความเร็วระดับ 100 เมกกะบิตต่อวินาที 36

4.28 การใช้ซีพียูของสวิตช์ขณะทำการทดสอบการรบกวนของระบบ 37

4.29 โครงสร้างไฟล์ที่ใช้ในการประกาศบริการฮาร์ดดิสก์ 38

4.30 ข้อความตอบกลับจากเครื่องที่ให้บริการฮาร์ดดิสก์ 38

4.31 ข้อความตอบกลับจากเครื่องที่ให้บริการฮาร์ดดิสก์ในกรณีที่มี 2 เครื่อง 38

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ข้อมูลต่างๆในปัจจุบันที่มุ่งสู่ยุคดิจิตอลได้เพิ่มขึ้นอย่างรวดเร็ว นำมาซึ่งความต้องการพื้นที่จัดเก็บข้อมูลอย่างมาก ความต้องการนี้ไม่เพียงจะเกิดขึ้นกับองค์กรขนาดใหญ่ องค์กรขนาดเล็กหรือผู้ใช้ตามบ้านเองก็มีความต้องการพื้นที่จัดเก็บข้อมูลมากขึ้นเช่นกัน การขยายพื้นที่จัดเก็บข้อมูลด้วยอุปกรณ์ที่มีความจุสูงจะมีราคาสูงตาม และอาจจะไม่เพียงพอกับความต้องการ ดังนั้นการขยายพื้นที่จัดเก็บข้อมูลด้วยการเพิ่มจำนวนอุปกรณ์ที่มีความจุต่อราคาที่เหมาะสมดูเหมือนจะเป็นทางเลือกที่เหมาะสมกว่า แต่การนำอุปกรณ์บันทึกข้อมูลหลายๆชิ้นมาเชื่อมต่อกันนั้น โดยทั่วไปการจัดเก็บข้อมูลจะกระจัดกระจายอยู่ตามที่ต่างๆและสามารถเก็บข้อมูลขนาดใหญ่ที่สุดได้ตามขนาดของอุปกรณ์ที่มีความจุสูงสุดเท่านั้น การตั้งค่าให้พื้นที่สำหรับจัดเก็บข้อมูลมีขนาดรวมกันเป็นเสมือนอุปกรณ์ขนาดใหญ่เพียงชิ้นเดียวนั้นค่อนข้างซับซ้อน และมีทางเลือกในแต่ละส่วนประกอบที่มีหลากหลาย

แม้ว่าการเพิ่มจำนวนอุปกรณ์จัดเก็บข้อมูลจะช่วยเพิ่มพื้นที่จัดเก็บข้อมูลได้ แต่การเพิ่มยังคงถูกจำกัดอยู่ที่พอร์ทเชื่อมต่อของอุปกรณ์ภายในของแต่ละเครื่อง ดังนั้นการใช้เครือข่ายคอมพิวเตอร์เข้ามาช่วยในการจัดเก็บข้อมูลจะช่วยลดข้อจำกัดนี้ลงได้ ระบบจัดเก็บข้อมูลแบบเครือข่ายหลักๆ แบ่งออกเป็นสองรูปแบบคือ Network Attached Storage (NAS) และ Storage Area Network (SAN)

Network Attached Storage เป็นระบบบันทึกข้อมูลที่มีเครื่องแม่ข่ายให้บริการพื้นที่บันทึกข้อมูลสำหรับลูกข่าย โดยที่ลูกข่ายสามารถเข้าถึงข้อมูลได้พร้อมๆกัน มีการจัดเก็บข้อมูลอยู่ที่ศูนย์กลางเพียงจุดเดียว ทำให้เครื่องลูกข่ายทุกเครื่องเป็นอิสระต่อกัน ถ้าเกิดเครื่องลูกข่ายเครื่องใดเครื่องหนึ่งมีปัญหาจะไม่กระทบกับการใช้งานของระบบโดยรวม และการที่มีเครื่องแม่ข่ายสำหรับการจัดเก็บข้อมูลเพียงจุดเดียวทำให้ง่ายต่อการบริหารจัดการทรัพยากรของพื้นที่บันทึกข้อมูล แต่ถ้าพื้นที่สำหรับการบันทึกข้อมูลไม่เพียงพอกับความต้องการ การขยายพื้นที่เพิ่มเติมโดยทั่วไปจำเป็นต้องปิดระบบก่อนการใส่อุปกรณ์เพิ่มเติม และมีข้อจำกัดของจำนวนฮาร์ดดิสก์ที่สามารถเพิ่มเติมได้ ซึ่งจะเกิดปัญหากับระบบที่จำเป็นต้องเรียกใช้ข้อมูลอยู่ตลอดเวลาและมีข้อมูลเป็นจำนวนมาก

Storage Area Network เป็นการนำอุปกรณ์บันทึกข้อมูลมาติดตั้งรวมกันเป็นเครือข่าย ให้บริการอุปกรณ์บันทึกข้อมูลโดยตรงกับเครื่องแม่ข่ายเสมือนเป็นอุปกรณ์บันทึกข้อมูลที่ต่อตรงของเครื่องแม่ข่ายเอง และการที่อุปกรณ์เหล่านี้ต่อเป็นเครือข่ายทำให้มีความสามารถในการต่อขยายเพิ่มเติมได้โดยไม่เกี่ยวกับพอร์ตของเครื่องแม่ข่าย แต่การใช้งานของระบบนี้อุปกรณ์ต่างๆ เป็นอุปกรณ์ที่เฉพาะเจาะจง มีการตั้งค่าที่ซับซ้อน และราคาที่สูง จึงไม่นิยมนำมาใช้งานทั่วไป นอกจากในองค์กรขนาดใหญ่เท่านั้น

การสร้างระบบจัดเก็บข้อมูลบนเครือข่ายโดยใช้เทคโนโลยีสำเร็จรูป เป็นอีกทางเลือกหนึ่งที่น่าสนใจเพราะเป็นเทคโนโลยีที่สามารถนำมาใช้ได้ง่าย ราคาถูก และมีความยืดหยุ่นสูงสามารถปรับแต่งได้อย่างเช่น อุปกรณ์สำหรับคอมพิวเตอร์ส่วนบุคคล หรือระบบปฏิบัติการลินุกซ์ ที่กำลังนิยมในปัจจุบัน แต่ด้วยความหลากหลายทำให้มีระบบไฟล์อยู่หลายระบบ การเลือกใช้ระบบไฟล์ที่เหมาะสมโดยมีประสิทธิภาพสูงสุดสำหรับงานแต่ละประเภทหรือโครงสร้างของเครือข่ายนั้นยังไม่สามารถระบุได้แน่ชัด

จากปัญหาที่กล่าวมา งานวิจัยนี้นำเสนอรูปแบบการสร้างระบบจัดเก็บข้อมูลบนเครือข่ายโดยใช้เทคโนโลยีสำเร็จรูป (Off-the-shelf Technology) ที่ง่ายต่อการใช้งาน สามารถขยายพื้นที่จัดเก็บข้อมูลได้ และเหมาะสมกับการใช้งานในประเภทต่างๆ โดยทำการทดสอบระบบด้วยตัวแปรต่างๆ เช่น รูปแบบการเชื่อมต่อ ระบบไฟล์ และโพรโทคอลในการรับส่งข้อมูล

1.2 วัตถุประสงค์ของการวิจัย

เพื่อนำเสนอรูปแบบของระบบจัดเก็บข้อมูลที่สามารถขยายพื้นที่จัดเก็บข้อมูลได้โดยอัตโนมัติ ต้นทุนต่ำ และมีประสิทธิภาพสูงเหมาะกับการใช้งานในลักษณะต่างๆ

1.3 ขอบเขตของการวิจัย

1.3.1 ใช้ชุดทดสอบเดียวกันในการทดสอบระบบรูปแบบต่างๆ

1.3.2 โพรโทคอลที่ใช้เชื่อมต่อฮาร์ดดิสก์ผ่านเครือข่ายจะใช้โพรโทคอลระดับบล็อกเท่านั้น

1.3.3 การวัดประสิทธิภาพจะใช้ค่าทรูพุต (Throughput)

1.3.4 จะทำการทดสอบโดยการรับ-ส่งข้อมูลผ่านเครื่องจริงในระบบปิด

1.3.5 เครือข่ายขนาดเล็กหมายถึงจำนวนการใช้งานพร้อมกันไม่เกิน 20 การเชื่อมต่อ

1.3.6 จำนวนเครื่องที่ให้บริการฮาร์ดดิสก์ที่มีในระบบมีจำนวนไม่เกิน 10 Node

1.3.7 การทดสอบระบบค้นหาและเชื่อมต่อฮาร์ดดิสก์ใช้วิธีการจำลอง

1.4 ประโยชน์ที่คาดว่าจะได้รับ

สามารถระบุส่วนประกอบในการสร้างระบบจัดเก็บข้อมูลที่เหมาะตามการใช้งานในรูปแบบต่างๆ โดยให้ประสิทธิภาพสูงสุด

1.5 วิธีดำเนินการวิจัย

1.5.1 ศึกษาเทคโนโลยีของระบบบันทึกข้อมูล

1.5.2 ศึกษาวิธีการในการค้นหาอุปกรณ์โดยอัตโนมัติ

1.5.3 ออกแบบขั้นตอนการทำงานและโครงสร้างของระบบ

1.5.4 ทดสอบวิธีการที่นำเสนอ

1.5.5 วิเคราะห์ผลการทดลอง

1.5.6 สรุปผลและเรียบเรียงวิทยานิพนธ์

1.6 ผลงานตีพิมพ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้ตีพิมพ์และนำเสนอในการประชุมวิชาการดังนี้

1.6.1 บทความชื่อ *"Building a Zero-Configuration Extensible Network Storage Based on Open Source Technology"* [1]

1.6.1.1 ชื่อผู้แต่ง Veerapat Phromchana Natawut Nupairoj และ Krerk Piromsopa

1.6.1.2 ตีพิมพ์และนำเสนอในงานประชุมวิชาการชื่อ *The 12th National Computer Science and Engineering Conference (NCSEC2008)* ซึ่งจัดขึ้นในวันที่ 20-21 พฤศจิกายน 2551 ณ จ.ชลบุรี ประเทศไทย

1.6.2 บทความชื่อ *"Performance evaluation of ZFS and LVM (with ext4) for scalable storage system"* [2]

1.6.2.1 ชื่อผู้แต่ง Veerapat Phromchana Natawut Nupairoj และ Krerk Piromsopa

1.6.2.2 ตีพิมพ์และนำเสนอในงานประชุมวิชาการชื่อ *The 8th International Joint Conference on Computer Science and Software Engineering (JCSSE2011)* ซึ่งจัดขึ้นในวันที่ 11-13 พฤษภาคม 2554 ณ จ.นครปฐม ประเทศไทย

บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

2.1 แนวคิดและทฤษฎี

งานวิจัยนี้ใช้ส่วนประกอบสำหรับการสร้างระบบจัดเก็บข้อมูลบนเครือข่าย ได้แก่ ระบบการค้นหาอุปกรณ์ โพรโทคอลเชื่อมต่ออุปกรณ์ผ่านเครือข่าย ระบบการจัดการฮาร์ดดิสก์ และระบบไฟล์ที่ขยายได้ โดยในแต่ละส่วนประกอบจะมีรายละเอียดดังนี้

2.1.1 การค้นหาอุปกรณ์

เพื่อช่วยอำนวยความสะดวกในการค้นหาบริการหรืออุปกรณ์ต่างๆ ที่ต่อพ่วงอยู่บนเครือข่าย เทคโนโลยีการค้นหาอุปกรณ์จึงเป็นแนวทางหนึ่งที่นิยมใช้กัน เทคนิคที่นิยมใช้กันสำหรับการค้นหาอุปกรณ์คือ Zeroconf [3]

Zeroconf เป็นเทคนิคที่ช่วยในการค้นหาบริการต่างๆบนเครือข่ายโดยอัตโนมัติ ไม่ต้องทราบหมายเลขไอพี หรือพอร์ทของเครื่องที่ให้บริการล่วงหน้า นอกจากนี้ Zeroconf ยังรองรับบริการต่างๆของสมาชิกในเครือข่ายที่เพิ่มขึ้นมา การทำงานของ Zeroconf แบ่งออกเป็น 3 ส่วนหลักๆคือ link-local addressing, multicast DNS และ DNS service discovery โดย link-local addressing จะทำหน้าที่ หาเลขไอพีและแปลงชื่อเครื่องเป็นเลขไอพี (ในกรณีที่ไม่มี DHCP หรือ DNS เซิร์ฟเวอร์ในเครือข่าย) จากนั้น multicast DNS จะทำหน้าที่ประกาศรายละเอียดของบริการต่างๆ บนระบบให้ผู้ใช้บนระบบเครือข่ายทราบ และส่วนสุดท้าย service discovery จะทำการค้นหาบริการต่างๆที่ให้บริการอยู่ในระบบ

จากคุณลักษณะของระบบ Zeroconf ดังกล่าว ทำให้ระบบ Zeroconf เหมาะสำหรับนำมาใช้เป็นโพรโทคอลในการค้นหาหน่วยเก็บข้อมูลที่เพิ่มเติมเข้ามาในระบบแบบอัตโนมัติ

2.1.2 การเชื่อมต่อฮาร์ดดิสก์ผ่านระบบเครือข่าย

ระบบเครือข่ายที่เก็บข้อมูล Storage Area Network (SAN) เป็นเครือข่ายสำหรับให้บริการที่เก็บข้อมูลในระดับบล็อก เสมือนเป็นฮาร์ดดิสก์ที่เชื่อมต่อโดยตรงเข้ากับตัวเครื่อง ซึ่งโพรโทคอลที่นิยมใช้ในฮาร์ดดิสก์สำหรับระบบ SAN ได้แก่ โพรโทคอล SCSI และได้มีการแปลงไปสู่โพรโทคอลต่างๆ เพื่อสร้างเป็นเครือข่ายเช่น iSCSI [4]

iSCSI (Internet Small Computer System Interface) เป็นโพรโทคอลมาตรฐานที่สามารถรับ-ส่งคำสั่ง SCSI บนเครือข่ายไอพี โดยจะทำการห่อหุ้มคำสั่งและข้อมูลในรูปของ iSCSI PDU (Protocol Data Unit) แล้วส่งให้ TCP layer สำหรับการส่งข้อมูลในเครือข่ายไอพี จากหลักการดังกล่าวทำให้ผู้ใช้ (Initiator) ที่ต้องการใช้งานอุปกรณ์บันทึกข้อมูลของเครื่องปลายทาง (Target) สามารถใช้งานได้โดยตรงในระดับ block unit I/O [5] เหมือนกับอุปกรณ์บันทึกข้อมูลที่อยู่ในตัวเครื่องคอมพิวเตอร์

นอกจากนี้ยังมีโพรโทคอลอีกชนิดหนึ่งที่ทำงานคล้ายกับ iSCSI คือโพรโทคอล AoE (ATA over Ethernet) [6] เป็นโพรโทคอลที่รับ-ส่งข้อมูล คำสั่ง ATA บนเครือข่ายเช่นเดียวกัน แต่โพรโทคอลนี้จะทำงานในระดับอีเทอร์เน็ต ซึ่งมีข้อดีคือภาระโหลดในการรับ-ส่งข้อมูลน้อย ทำให้การรับ-ส่งข้อมูลแต่ละครั้งมีความกระชับ แต่ข้อเสียคือไม่สามารถใช้เครื่องมือจัดการเครือข่ายที่แพร่หลายในปัจจุบันได้

ด้วย iSCSI สามารถใช้งานบนเครือข่ายทั่วไปได้และง่ายต่อการจัดการ จึงทำให้มีความเหมาะสมที่จะนำมาใช้งานมากกว่า อย่างไรก็ตามเพื่อเชื่อมต่อฮาร์ดดิสก์ทั่วไปเข้ากับ iSCSI จะต้องอาศัยเครื่องคอมพิวเตอร์หรืออุปกรณ์เพื่อทำหน้าที่ให้บริการ iSCSI เพิ่มเติมด้วยโดยจะมีการเรียกชื่อฮาร์ดดิสก์ตามมาตรฐาน iqn [7]

2.1.3 **การจัดการพื้นที่เก็บข้อมูล** (Volume Manager)

พื้นที่บนอุปกรณ์บันทึกข้อมูลโดยปกติมีขนาดเท่ากับพื้นที่ของอุปกรณ์นั้นๆ การใช้งานจะอ้างอิงกับพื้นที่จริงของอุปกรณ์ ทำให้อุปกรณ์แต่ละชิ้นเป็นอิสระต่อกัน การนำมาใช้งานร่วมกันจึงมีความยุ่งยาก จำเป็นที่จะต้องมีตัวจัดการให้เป็นพื้นที่เสมือนเพื่อให้ง่ายในการอ้างอิงใช้งาน

Logical Volume Manager (LVM) [8] เป็นตัวจัดการพื้นที่บนอุปกรณ์บันทึกข้อมูล ที่มีความยืดหยุ่นมากกว่าการแบ่งพาร์ติชันแบบธรรมดา โดยมีความสามารถในการนำพาร์ทิชั่น (physical volume) หลายๆส่วนรวมเข้าด้วยกันเป็นกลุ่ม (volume groups) ให้เป็นเหมือนพื้นที่เสมือน (logical volume) ขนาดใหญ่ ที่สามารถเปลี่ยนแปลงขนาดหรือเคลื่อนย้ายได้ในขณะที่กำลังใช้งาน

2.1.4 ระบบไฟล์ที่ปรับเปลี่ยนขนาดได้

การจัดเก็บข้อมูลในระบบที่สามารถขยายพื้นที่จัดเก็บได้นั้น จำเป็นต้องมีระบบไฟล์ที่ยืดหยุ่นสามารถขยายขนาดตามพื้นที่บันทึกข้อมูลที่เพิ่มขึ้นได้ ระบบไฟล์ที่นิยมใช้ในปัจจุบันได้แก่ ext3 [9], ext4 [10], NTFS [11], ReiserFS [12] และ ZFS [13]

ext3, ext4 เป็นระบบไฟล์ที่ใช้กันทั่วไปสำหรับระบบปฏิบัติการลินุกซ์ สามารถขยายขนาดของระบบไฟล์ได้ตามขนาดของพาร์ทิชันที่ใหญ่ขึ้น และในปัจจุบันมีเครื่องมือที่สามารถขยายขนาดของระบบไฟล์ ได้ขณะที่กำลังใช้งาน

NTFS เป็นระบบไฟล์มาตรฐานของระบบปฏิบัติการวินโดวส์ การนำมาใช้ในระบบปฏิบัติการลินุกซ์นั้นยังไม่สามารถใช้งานได้อย่างเต็มประสิทธิภาพ

ReiserFS เป็นระบบไฟล์ในระบบปฏิบัติการลินุกซ์ ที่พัฒนาโดยบริษัท Namesys มีความสามารถที่จะขยายขนาดของระบบไฟล์ได้ในขณะที่กำลังใช้งาน

ZFS เป็นระบบไฟล์ที่รวมส่วนจัดการพื้นที่เก็บข้อมูลไว้ด้วยกัน พัฒนาโดยบริษัทซัน ไมโครซิสเต็มส์ ปัจจุบันได้มีการแปลงไปใช้ในระบบปฏิบัติการต่างๆ เช่น โซลาริส ฟรีบีเอสดี และลินุกซ์

2.2 เอกสารและงานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับระบบจัดการข้อมูลบนเครือข่ายสามารถแบ่งได้เป็น 2 ประเภทคือ

2.2.1 งานวิจัยที่เกี่ยวข้องทางด้านระบบจัดเก็บข้อมูลที่ขยายพื้นที่จัดเก็บได้

งานวิจัย [14] ของ Yasuda และคณะ นำเสนอแนวคิดให้เครื่องจัดเก็บข้อมูลบนเครือข่ายทั่วไปสามารถต่อพ่วงกันได้เพื่อเพิ่มพื้นที่ใช้งาน โดยกำหนดให้เป็นเครื่องแม่ข่ายหนึ่งเครื่อง ทำหน้าที่เชื่อมต่อพื้นที่ใช้งานของเครื่องจัดเก็บข้อมูลอื่นแล้วรวมให้เสมือนเป็นพื้นที่เดียวกันขนาดใหญ่ และส่งต่อข้อมูลไปยังเครื่องจัดเก็บข้อมูลในเครือข่ายในระดับไฟล์

งานวิจัย [15] ของ Qingguo และคณะ นำเสนอการสร้างระบบจัดเก็บข้อมูลที่สามารถขยายพื้นที่จัดเก็บได้ด้วยซอฟต์แวร์โอเพนซอร์ส ระบบประกอบด้วยเครื่องแม่ข่ายทำหน้าที่เชื่อมต่อกับตัวเก็บข้อมูลผ่านทางระบบเครือข่ายในระดับบล็อก และให้บริการพื้นที่เก็บข้อมูลแก่เครื่องลูกข่าย เครื่องแม่ข่ายสามารถขยายพื้นที่เก็บข้อมูลได้เรื่อยๆ โดยการเพิ่มตัวเก็บข้อมูลเข้าไปในระบบ

2.2.2 งานวิจัยที่เกี่ยวข้องทางด้านการวัดประสิทธิภาพโพรโทคอลในการรับ-ส่งข้อมูล

งานวิจัย [16] ของ Yingping และ David ได้ศึกษาประสิทธิภาพของโพรโทคอล iSCSI โดยมีการเปรียบเทียบกับระบบ SAN และโพรโทคอลที่ใช้สำหรับแชร์ข้อมูลในระดับไฟล์เช่น NFS ในยูนิกซ์ และ CIFS ในวินโดวส์ พบว่าโพรโทคอลในระดับบล็อกรวมถึง iSCSI ให้ประสิทธิภาพที่สูงกว่าโพรโทคอลในระดับไฟล์

บทที่ 3

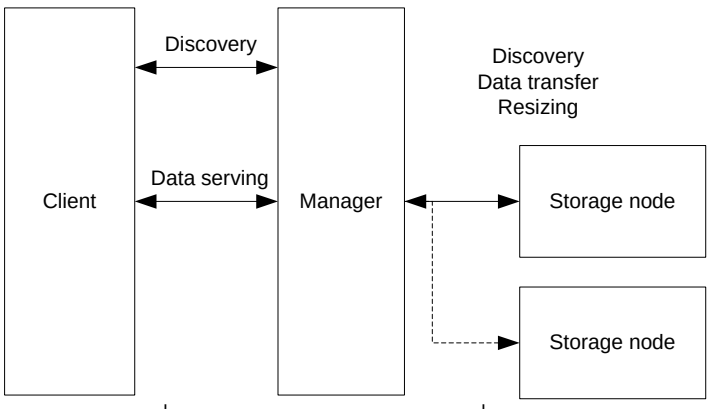
การออกแบบ

จากงานวิจัยที่เกี่ยวข้องต่างๆ นำเสนอการสร้างระบบจัดเก็บข้อมูลที่สามารถขยายพื้นที่จัดเก็บข้อมูลได้ด้วยการใช้เครื่องจัดการขั้นกลางระหว่างผู้ใช้ และหน่วยจัดเก็บข้อมูล แต่ยังไม่มีระบบใดที่สามารถเพิ่มหน่วยจัดเก็บข้อมูลแล้วสามารถใช้งานได้โดยอัตโนมัติ ผู้ใช้จำเป็นต้องตั้งค่าต่างๆ ด้วยตัวเอง

งานวิจัยนี้จึงได้นำเสนอระบบจัดเก็บข้อมูลแบบขยายพื้นที่จัดเก็บได้โดยอัตโนมัติ ซึ่งแบ่งออกเป็นสองส่วนคือแบบจำลองของระบบจัดเก็บข้อมูลแบบขยายพื้นที่จัดเก็บได้โดยอัตโนมัติ และการวัดประสิทธิภาพส่วนประกอบของระบบตามการใช้งาน

3.1 แบบจำลองระบบจัดเก็บข้อมูลแบบขยายพื้นที่จัดเก็บข้อมูลโดยอัตโนมัติ

ในส่วนของระบบจัดเก็บข้อมูลแบบขยายพื้นที่จัดเก็บข้อมูลได้โดยอัตโนมัตินั้น งานวิจัยนี้มีแนวคิดที่จะใช้เครื่องจัดการ (Manager) เป็นเครื่องแม่ข่ายในการให้บริการพื้นที่บันทึกข้อมูลแก่เครื่องลูกข่าย (Client) และทำหน้าที่ค้นหาเครื่องที่ให้บริการฮาร์ดดิสก์ (Storage Node) โดยทำการเชื่อมต่อฮาร์ดดิสก์เข้ากับเครื่องจัดการและเตรียมให้พร้อมใช้งานโดยอัตโนมัติ ดังรูปที่ 3.1



รูปที่ 3.1 แบบจำลองของระบบที่ออกแบบ

เครื่องคอมพิวเตอร์ในระบบที่ออกแบบจะใช้ระบบปฏิบัติการลินุกซ์ เนื่องจากสามารถปรับแต่งได้ง่าย และมีส่วนโปรแกรมอรรถประโยชน์ต่างๆ ที่พร้อมใช้อยู่มากมาย

3.1.1 การตรวจหาอุปกรณ์ใหม่

ในการตรวจหาอุปกรณ์ใหม่ที่เพิ่มเข้ามาในระบบ เครื่องคอมพิวเตอร์ทุกเครื่องต้องมีโปรแกรม Avahi [17] ซึ่งส่วนของเครื่องจัดการจะใช้ส่วนของ DNS Service Discovery ในการค้นหาอุปกรณ์ และเครื่องที่ให้บริการฮาร์ดดิสก์จะใช้ส่วนของ mDNS Responder ที่ให้รายละเอียดต่างๆ เช่น ชื่อเครื่อง หมายเลขไอพี โพรโทคอล และชื่อฮาร์ดดิสก์ที่ให้บริการ เมื่อมีการร้องขอจากเครื่องจัดการ

3.1.2 การเชื่อมต่อกับฮาร์ดดิสก์

การเชื่อมต่อกับฮาร์ดดิสก์ จะเน้นการเชื่อมต่อที่สามารถใช้งานผ่านเครือข่ายได้ เนื่องจากมีความยืดหยุ่นในการเพิ่มฮาร์ดดิสก์กว่าการเชื่อมต่อโดยตรงเข้ากับตัวเครื่องแม่ข่าย โดยเลือกใช้โพรโทคอลที่มีประสิทธิภาพสูงในการรับ-ส่งข้อมูล เน้นโพรโทคอลในระดับบล็อกเนื่องจากบริหารจัดการได้ง่าย และมีประสิทธิภาพสูงเนื่องจากสามารถเข้าสู่พื้นที่เก็บข้อมูลได้โดยตรง เช่น AoE และ iSCSI

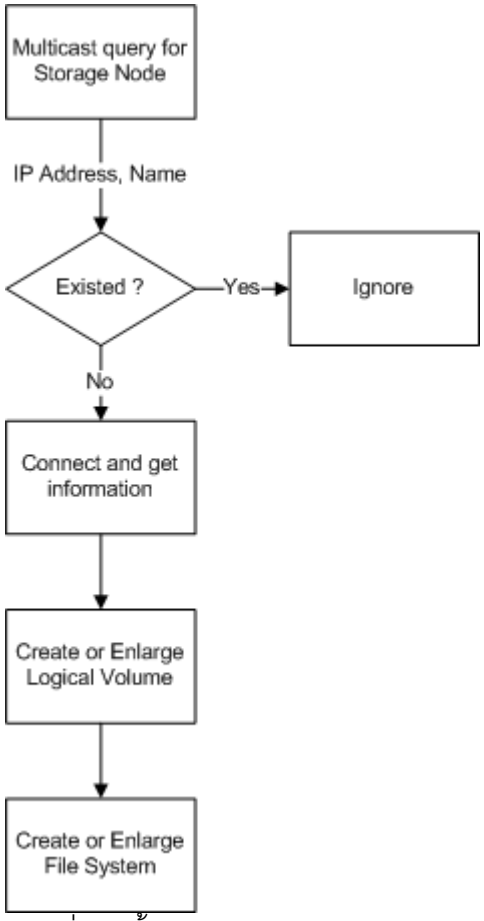
3.1.3 การจัดการพื้นที่สำหรับผู้ใช้

เมื่อทำการเชื่อมต่อฮาร์ดดิสก์ผ่านเครือข่ายเข้าสู่เครื่องจัดการแล้ว ในระบบปฏิบัติการลินุกซ์จะมี LVM เป็นตัวจัดการพาร์ติชันที่สามารถนำพาร์ติชันมาต่อเรียงกันเป็นพาร์ติชันเสมือนขนาดใหญ่ และสามารถปรับเปลี่ยนขนาดพาร์ติชันภายหลังได้ และงานวิจัยนี้จะเลือกใช้ระบบไฟล์ที่สนันสนุนคุณสมบัติการขยายขนาดระบบไฟล์ขณะที่กำลังใช้งานอยู่ได้ ทำให้ผู้ใช้ยังคงสามารถใช้งานได้อย่างต่อเนื่องไม่ได้กระทบกับค่าที่ตั้งไว้ก่อนหน้า โดยระบบไฟล์ที่มีคุณสมบัติดังกล่าวได้แก่ ext3, ext4, ReiserFS และ ZFS

3.1.4 ขั้นตอนการทำงานของเครื่องจัดการ

เครื่องจัดการจะส่งข้อความมัลติคาสต์ไปในระบบเครือข่ายเพื่อค้นหาเครื่องที่ให้บริการฮาร์ดดิสก์ที่อยู่ในระบบเป็นระยะ เมื่อเครื่องที่ให้บริการฮาร์ดดิสก์ได้รับข้อความจากเครื่องจัดการก็จะทำการตอบกลับด้วยรายละเอียดต่างๆ เช่นหมายเลขไอพี รายชื่อฮาร์ดดิสก์ โพรโทคอลที่ใช้เชื่อมต่อ เครื่องจัดการจะทำการตรวจสอบกับฐานข้อมูลว่าชื่อฮาร์ดดิสก์ที่เครื่องให้บริการตอบกลับมานั้นมีอยู่ก่อนแล้วในระบบหรือไม่ ถ้ามีอยู่ก่อนแล้วก็จะไม่สนใจ แต่ถ้ายังไม่มีก็จะทำการ

เชื่อมต่อเข้ากับเครื่องจัดการ ขยายขนาดพื้นที่เสมือน และขยายขนาดระบบไฟล์ให้ครอบคลุมพื้นที่บนฮาร์ดดิสก์ใหม่ที่เพิ่มเข้ามา ดังรูปที่ 3.2



รูปที่ 3.2 ขั้นตอนการทำงานของระบบ

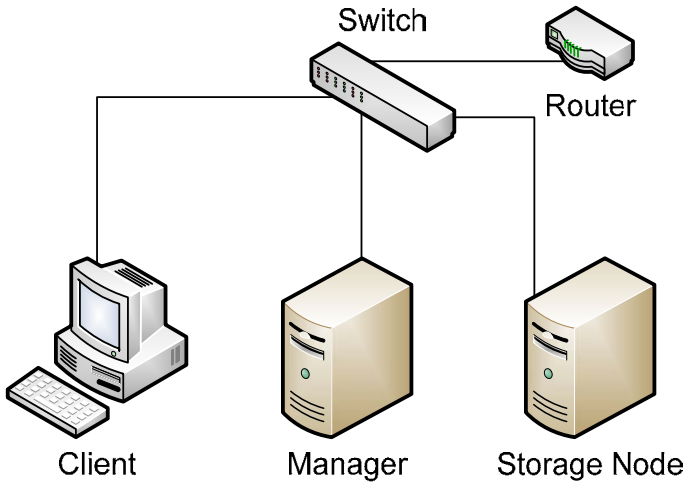
3.2 การวัดประสิทธิภาพส่วนประกอบของระบบ

3.2.1 การทดลอง

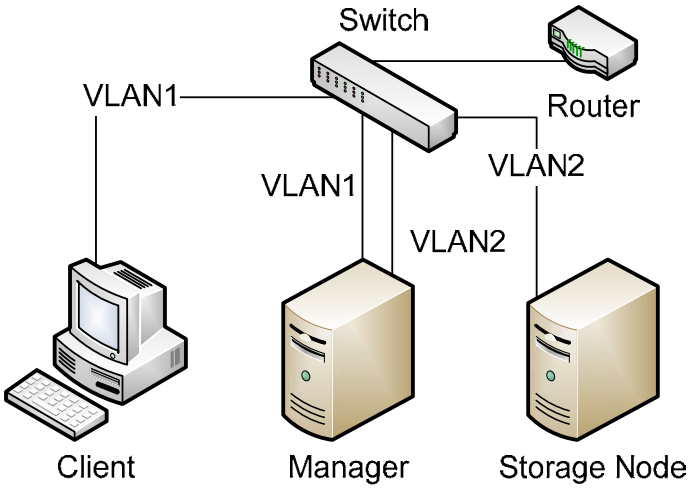
ในการทดสอบวัดประสิทธิภาพส่วนประกอบจะทำการทดสอบโดยการตั้งค่าต่างๆดังตารางที่ 3.1 ระบบประกอบไปด้วยเครื่องคอมพิวเตอร์ 3 เครื่อง เชื่อมต่อกับระบบเครือข่ายใน 2 รูปแบบ ได้แก่ เครือข่ายที่รวมส่วนจัดเก็บข้อมูล (LAN-like model) และเครือข่ายที่แยกส่วนจัดเก็บข้อมูล (SAN-like model) ดังรูปที่ 3.3 เครือข่ายแบบรวมส่วนจัดเก็บข้อมูลรูปที่ 3.3 และรูปที่ 3.4 ตามลำดับ

ตารางที่ 3.1 รายละเอียดของส่วนประกอบต่างๆที่ทำการทดสอบ

จำนวนการใช้งานพร้อมกัน	รูปแบบการเชื่อมต่อเครือข่าย	ความเร็วเครือข่าย	การเข้าถึงส่วนจัดเก็บข้อมูล	ระบบไฟล์
1 5 10 15 20	รวมส่วนจัดเก็บข้อมูล แยกส่วนจัดเก็บข้อมูล	100 Mbps (FE) 1 Gbps (GbE)	iSCSI AoE	ext3 ext4 ReiserFS ZFS

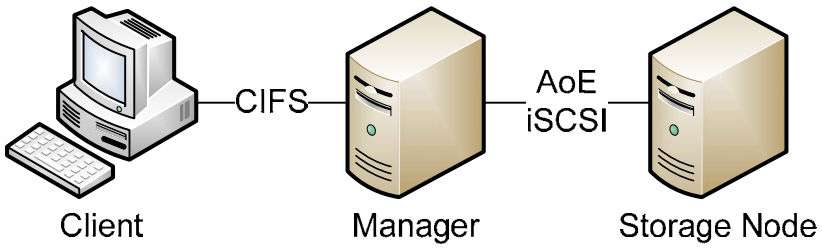


รูปที่ 3.3 เครือข่ายแบบรวมส่วนจัดเก็บข้อมูล (LAN-like model)



รูปที่ 3.4 เครือข่ายแบบแยกส่วนจัดเก็บข้อมูล (SAN-like model)

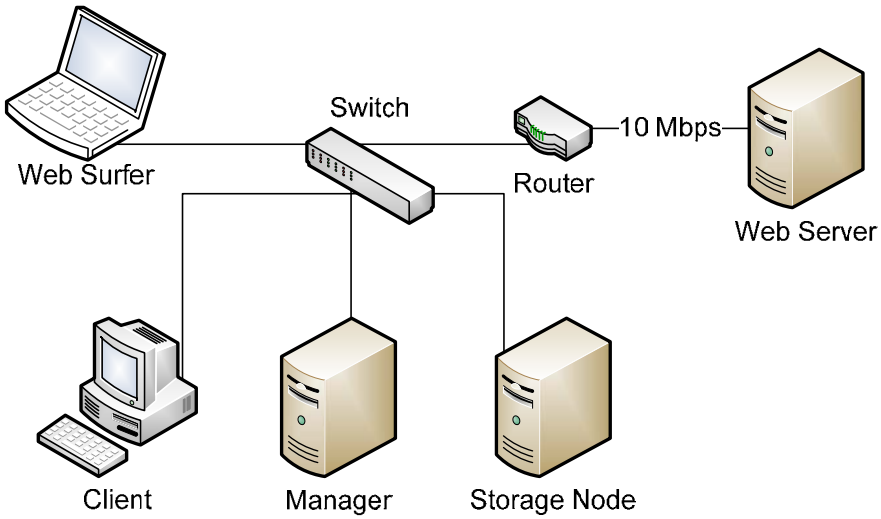
การวัดประสิทธิภาพใช้วิธีวัดความเร็วในการรับ-ส่งข้อมูลจากเครื่องลูกข่ายผ่านทางโพรโทคอล CIFS [18] ซึ่งเป็นโพรโตคอลมาตรฐานที่ใช้ในการแชร์ไฟล์บนระบบปฏิบัติการวินโดวส์ และรองรับการใช้งานในระบบปฏิบัติการลินุกซ์ ไปยังเครื่องจัดการดังรูปที่ 3.5 โดยจำลองตามการใช้งาน เช่น จำนวนการใช้งานพร้อมกัน ขนาดของไฟล์ และความเร็วของเครือข่าย ด้วยโปรแกรม IOzone [19] บนระบบปฏิบัติการลินุกซ์ซึ่งเป็นโปรแกรมที่สามารถทดสอบไฟล์ในขนาดต่างๆที่ครอบคลุมถึงไฟล์ทั่วไปในปัจจุบันได้ดี



รูปที่ 3.5 โพรโทคอลที่ใช้ในการทดสอบส่วนประกอบ

3.2.1 ปัจจัยภายนอก

ในการใช้การโดยทั่วไปปกติจะมีการใช้งานอย่างอื่นนอกเหนือจากพื้นที่จัดเก็บข้อมูลเพียงอย่างเดียว เช่น การใช้งานอินเตอร์เน็ท เข้าเว็บไซต์ เช็คอีเมล์ หรือดาวน์โหลดข้อมูลต่างๆด้วย งานวิจัยนี้จะทำการจำลองกระแสข้อมูลขนาด 10 เมกกะบิตต่อวินาที ซึ่งเป็นความเร็วอินเตอร์เน็ทสูงสุดสำหรับผู้ใช้ตามบ้าน หรือบริษัทขนาดเล็ก ที่มีในท้องตลาดขณะที่ทำการวิจัย พร้อมกับการใช้งานพื้นที่จัดเก็บข้อมูล เพื่อดูผลกระทบต่อประสิทธิภาพในการใช้งาน การรบกวนนี้จะเกิดขึ้นจากเครื่องลูกข่ายที่ทำการทดสอบเอง และเครื่องอื่นที่อยู่ในเครือข่ายเดียวกัน ดังรูปที่ 3.6



รูปที่ 3.6 การทดสอบการรบกวนของระบบ

การทดสอบปัจจัยภายนอกจะเปลี่ยนโปรแกรมทดสอบเป็น Intel NAS Performance Toolkit [20] (NASPT) ซึ่งทำงานบนระบบปฏิบัติการวินโดวส์ เอ็กซ์พี โปรแกรมนี้จะทดสอบบนพื้นฐานของการใช้งานจริงในด้านต่างๆเช่น การเล่นวีดิโอความละเอียดสูง, การเล่นวีดิโอความละเอียดสูงพร้อมกัน 2 และ 4 ไฟล์, การบันทึกวีดิโอความละเอียดสูง, การเล่นวีดิโอความละเอียดสูงพร้อมกับบันทึกวีดิโอ, การสร้างวีดิโอและรูปภาพ, การใช้งานทั่วไปในสำนักงาน, การคัดลอกข้อมูลไป-กลับ, การคัดลอกไดเรกทอรีไป-กลับ และการชมอัลบั้มรูป โดยในการทดสอบเครื่องลูกข่ายจะทำการดาวน์โหลดข้อมูลจากเครื่องบริการเว็บที่อยู่นอกเครือข่ายที่ใช้ทำการทดสอบ เพื่อที่จะให้กระแสข้อมูลนี้ครอบคลุมทุกการทดสอบ เครื่องลูกข่ายจะทำการดาวน์โหลดข้อมูลขนาด 40 กิกะไบต์ซึ่งใช้เวลาในการดาวน์โหลดเร็วที่สุดอย่างน้อย 10 ชั่วโมง ซึ่งมากพอที่จะทำการทดสอบในแต่ละเงื่อนไขโดยไม่มีการหยุดของกระแสข้อมูลก่อนเสร็จสิ้นการทดสอบ

3.2.2 **อุปกรณ์ที่ใช้ในการทดสอบ**

อุปกรณ์ที่ใช้ในการทดสอบจะประกอบไปด้วย เครื่องคอมพิวเตอร์ 3 เครื่อง แบ่งเป็นเครื่องลูกข่าย เครื่องจัดการ และเครื่องที่ให้บริการฮาร์ดดิสก์ กิกะบิตสวิตช์ และเราเตอร์ ดังตารางที่ 3.2

ตารางที่ 3.2 รายละเอียดอุปกรณ์ที่ใช้ในการทดสอบ

	Client	Manager	Storage Node
Processor	Intel Pentium 4 2.8 GHz	AMD Athlon X2 3600+	
Memory	1024 MB		
Storage	Silicon Power SSD 64GB	Seagate ST3250310AS x4	
NIC(s)	1GbE	1GbE x2	1GbE
OS	Linux 2.6.32, Windows XP	Linux 2.6.32 AMD64	Linux 2.6.32
Switch	DELL PowerConnect 2808		
Router	Cisco Linksys WRT110		

เครือข่ายที่ทำการทดสอบจะไม่มีเครื่องคอมพิวเตอร์จากภายนอกเข้ามาเกี่ยวข้อง และค่ามาตรฐานของเราเตอร์จะเปิดไฟร์วอลล์เพื่อปิดกันการเชื่อมต่อจากภายนอก กล่าวคือ เครือข่ายที่ทำการทดสอบจะไม่มีการรบกวนจากภายนอก

3.2.3 การทดสอบการทำงานของแบบจำลอง

การทดสอบการทำงานของแบบจำลองประกอบไปด้วยขั้นตอน การค้นหาเครื่องที่ให้บริการฮาร์ดดิสก์ การเชื่อมต่อ การสอบถามขนาดของฮาร์ดดิสก์ การขยายขนาดพื้นที่เสมือนและการขยายขนาดของระบบไฟล์ การทดสอบทำอยู่บนเครื่องคอมพิวเตอร์เสมือนผ่านโปรแกรมจำลอง VMware [21] โดยเพิ่มเครื่องที่ให้บริการฮาร์ดดิสก์เข้าไปในระบบทีละเครื่อง หรือครั้งละหลายเครื่องด้วยจำนวนสุ่ม การทดสอบนี้วัดผลโดยดูจากขนาดของพื้นที่จัดเก็บข้อมูลรวมว่ามีขนาดเท่ากับฮาร์ดดิสก์ที่มีทั้งหมดในระบบหรือไม่

บทที่ 4

การทดสอบ

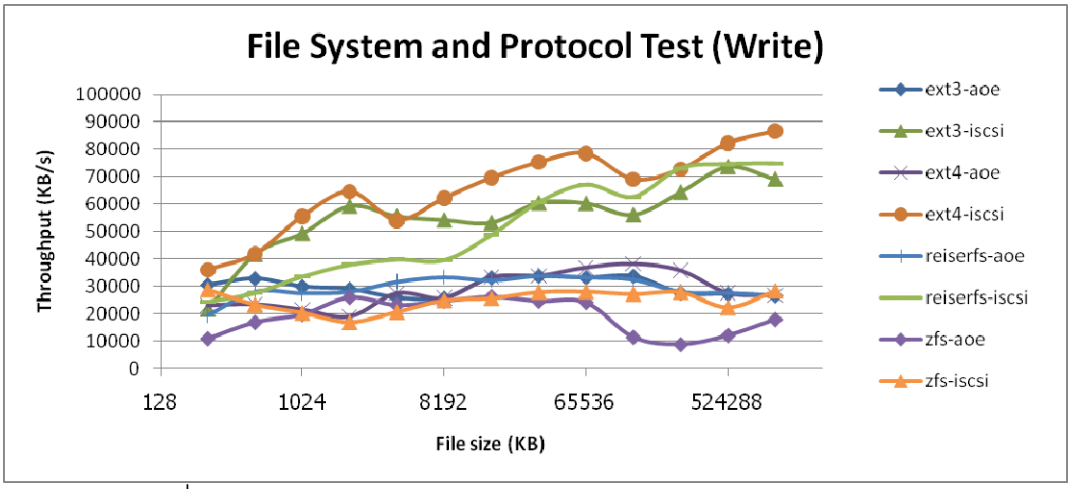
การทดสอบระบบที่ออกแบบทำโดยการวัดประสิทธิภาพส่วนประกอบต่างๆตามโครงสร้างของแบบจำลอง เพื่อหาส่วนประกอบที่เหมาะสมกับการใช้งานประเภทต่างๆ จากนั้นจะทำการทดสอบการทำงานของระบบเชิงหน้าที่ว่าสามารถทำงานได้ถูกต้องหรือไม่

4.1 การวัดประสิทธิภาพส่วนประกอบของระบบ

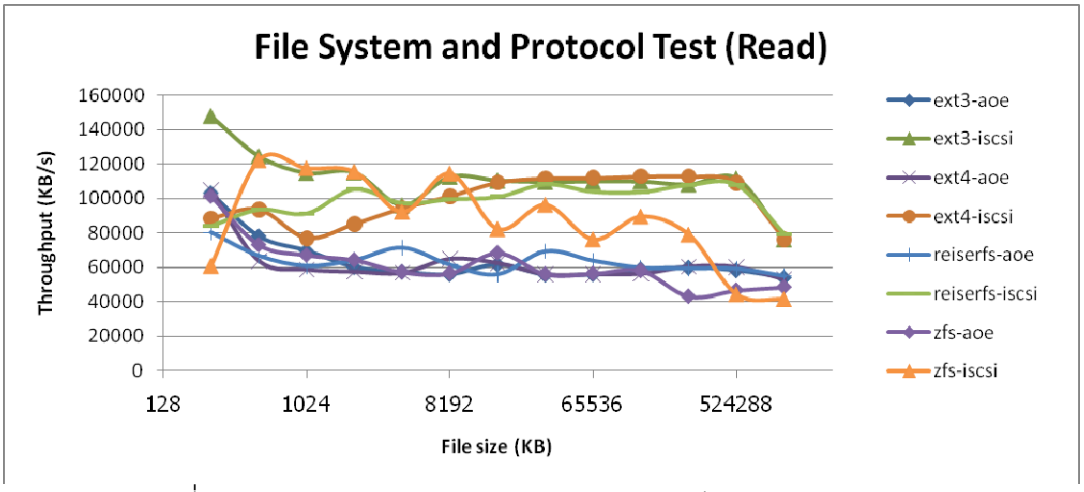
4.1.1 ผลการทดลอง

การทดลองเบื้องต้นจะทำการวัดประสิทธิภาพระหว่างระบบไฟล์และโพรโทคอลที่ใช้ในการเชื่อมต่อฮาร์ดดิสก์ผ่านเครือข่าย โดยทำการทดสอบระหว่างเครื่องจัดการและเครื่องที่ให้บริการฮาร์ดดิสก์ เพื่อคัดเลือกส่วนประกอบพื้นฐานที่มีประสิทธิภาพสูงมาวิเคราะห์ในเงื่อนไขอื่นๆต่อไป

ผลการทดสอบเบื้องต้นในรูปที่ 4.1 และรูปที่ 4.2 พบว่าระบบไฟล์ ext3 ext4 และ ReiserFS บนโพรโทคอล iSCSI ให้ประสิทธิภาพสูงที่ขนาดไฟล์ต่างๆอย่างเด่นชัดทั้งการเขียนและอ่านข้อมูล ดังนั้นการทดสอบเงื่อนไขอื่นๆที่เหลือจะทำการทดลองบนพื้นฐานส่วนประกอบ 3 เงื่อนไขนี้



รูปที่ 4.1 ความเร็วในการเขียนข้อมูลระหว่างระบบไฟล์และโพรโทคอล



รูปที่ 4.2 ความเร็วในการอ่านข้อมูลระหว่างระบบไฟล์และโพรโทคอล

การทดสอบเงื่อนไขอื่นๆที่เหลือจะทำการทดสอบโดยใช้ขนาดไฟล์ 3 ขนาด ได้แก่ 1, 4 และ 350 เมกกะไบต์ เป็นตัวแทนของไฟล์เอกสาร รูปภาพ เพลง และไฟล์ขนาดใหญ่อื่นๆ ทำการทดสอบเขียน-อ่านพร้อมกัน 1, 5, 10, 15 และ 20 ไฟล์พร้อมกัน บนเครือข่ายที่เชื่อมต่อ 2 แบบ ทั้ง 2 ความเร็ว

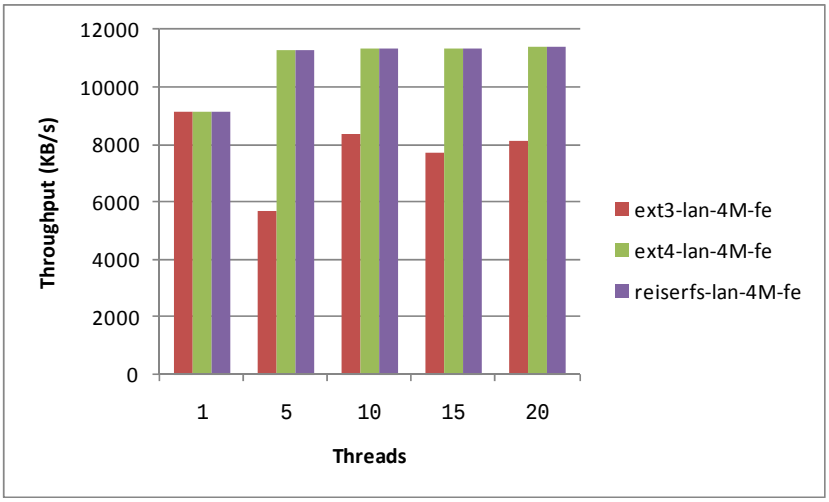
จากผลการทดลองวัดความเร็วการรับ-ส่งข้อมูลในเครือข่ายแบบรวมส่วนเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาทีดังตารางที่ 4.1 และตารางที่ 4.2 ความเร็วที่ได้นั้นค่อนข้างใกล้เคียงกันยกเว้นการอ่านข้อมูลขนาด 4 เมกกะไบต์ ระบบไฟล์ ext3 จะอ่านข้อมูลได้ช้ากว่าระบบไฟล์อื่นในขณะที่มีการใช้งานพร้อมกันตั้งแต่ 5 ไฟล์ขึ้นไปดังรูปที่ 4.3 และการเขียนข้อมูลที่ขนาด 350 เมกกะไบต์ ระบบไฟล์ ext4 มีความเร็วต่ำสุดประมาณ 8 เมกกะไบต์ต่อวินาที ที่มีการใช้งานพร้อมกันตั้งแต่ 5 ไฟล์ขึ้นไปดังรูปที่ 4.4 แต่ในทางกลับกัน การอ่านข้อมูลกลับมีความเร็วสูงสุดในการใช้งานพร้อมกันตั้งแต่ 10 ไฟล์ขึ้นไป ส่วนระบบไฟล์ ReiserFS ข้อมูลขนาด 350 เมกกะไบต์ ความเร็วการเขียนไฟล์ทำได้ในระดับปานกลาง และต่ำสุดในการอ่านไฟล์ สำหรับการใช้งานพร้อมกันตั้งแต่ 5 ไฟล์ขึ้นไป

ตารางที่ 4.1 ผลการวัดประสิทธิภาพการเขียนข้อมูลในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที)

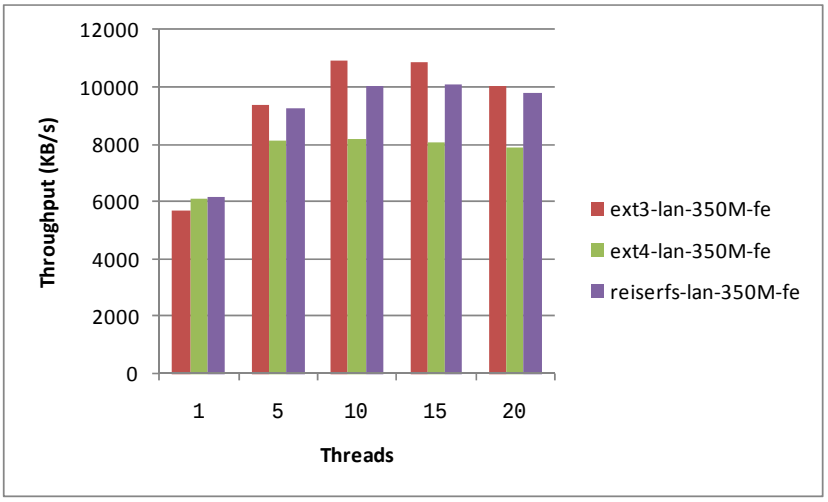
File system	File size	Threads				
		1	5	10	15	20
ext3	1 MB	9566.79	10885.18	10814.19	10800.94	10785.11
	4 MB	9487.27	11318.01	11071.63	11001.34	11144.81
	350 MB	5704.54	9398.26	10933.59	10858.06	10008.31
ext4	1 MB	9797.08	10818.24	10818.14	10799.28	10787.24
	4 MB	9831.81	11319.73	11304.04	11298.96	11297.45
	350 MB	6124.16	8149.12	8188.87	8045.75	7859.35
ReiserFS	1 MB	9542.72	10880.98	9977.72	10795.27	10653.07
	4 MB	9432.64	11319.76	11304.52	11300.17	11222.38
	350 MB	6172.95	9280.76	10004.97	10076.36	9769.93

ตารางที่ 4.2 ผลการวัดประสิทธิภาพการอ่านข้อมูลในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที)

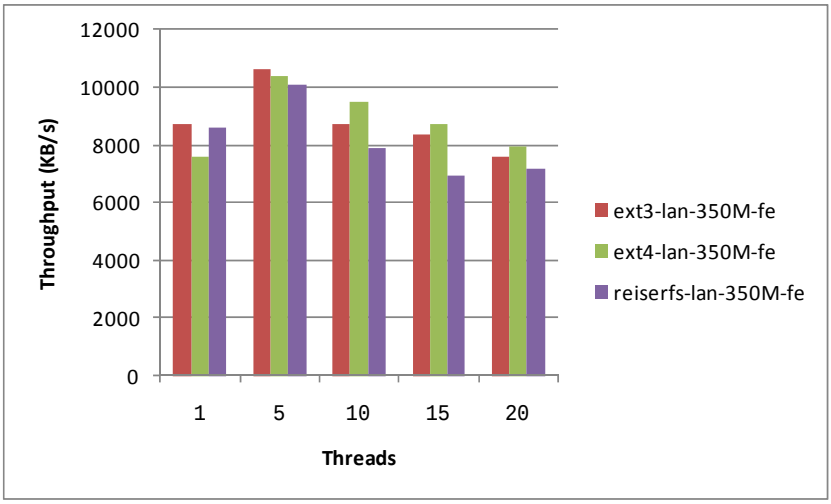
File system	File size	Threads				
		1	5	10	15	20
ext3	1 MB	9036.74	10850.89	11119.77	10894.70	9489.18
	4 MB	9134.72	5687.70	8358.89	7723.72	8094.62
	350 MB	8704.57	10597.41	8742.91	8373.56	7585.48
ext4	1 MB	9036.76	10826.31	11119.24	11406.97	11266.44
	4 MB	9129.75	11257.37	11333.27	11358.97	11372.12
	350 MB	7612.80	10400.98	9469.91	8733.30	7925.82
ReiserFS	1 MB	9076.73	10827.79	8309.60	11217.47	11264.51
	4 MB	9140.40	11265.41	11337.86	11340.72	11367.44
	350 MB	8593.43	10068.82	7901.64	6935.71	7190.10



รูปที่ 4.3 ความเร็วในการอ่านไฟล์ขนาด 4 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที



รูปที่ 4.4 ความเร็วในการเขียนไฟล์ขนาด 350 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที



รูปที่ 4.5 ความเร็วในการอ่านไฟล์ขนาด 350 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที

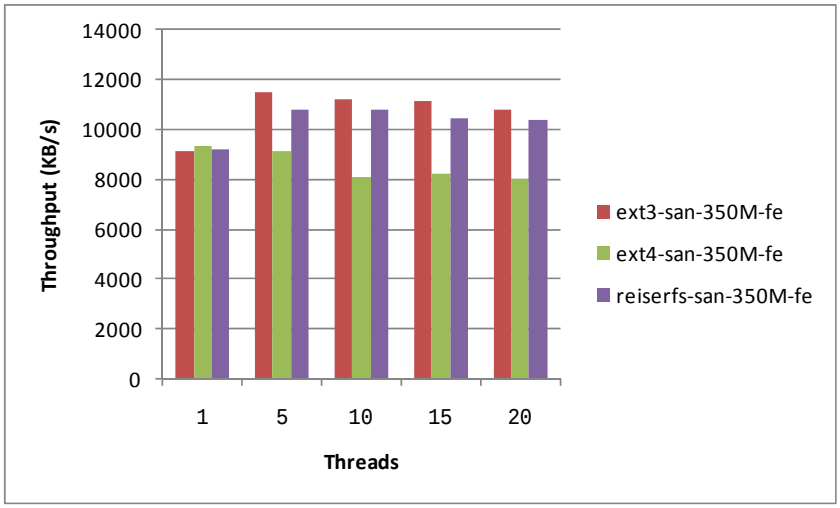
ส่วนผลการทดลองในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาทีดังตารางที่ 4.3 และตารางที่ 4.4 ความเร็วที่ได้ทั้งการเขียน-อ่านข้อมูลค่อนข้างจะใกล้เคียงกันในระบบไฟล์ต่างๆที่ขนาดไฟล์ 1 และ 4 เมกกะไบต์ ส่วนไฟล์ขนาด 350 เมกกะไบต์ระบบไฟล์ ext3 มีความเร็วสูงสุดทั้งการเขียนและอ่านไฟล์ ดังรูปที่ 4.6 และรูปที่ 4.7 ระบบไฟล์ ext4 ยังคงมีความเร็วต่ำที่สุดในการเขียนข้อมูลพร้อมกันตั้งแต่ 5 ไฟล์ขึ้นไป โดยความเร็วจะอยู่ประมาณ 8 เมกกะไบต์ต่อวินาทีเหมือนกับเครือข่ายที่รวมส่วนจัดเก็บข้อมูลในการทดลองที่แล้ว ส่วนความเร็วในการอ่านข้อมูลมีความเร็วใกล้เคียงกับระบบไฟล์ ext3 ระบบไฟล์ ReiserFS ทำความเร็วในการเขียนไฟล์ได้ค่อนข้างดีช้ากว่าระบบไฟล์ ext3 เล็กน้อย แต่ความเร็วในการอ่านข้อมูลนั้นต่ำที่สุดในกลุ่มระบบไฟล์ที่ทดสอบ

ตารางที่ 4.3 ผลการวัดประสิทธิภาพการเขียนข้อมูลในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที)

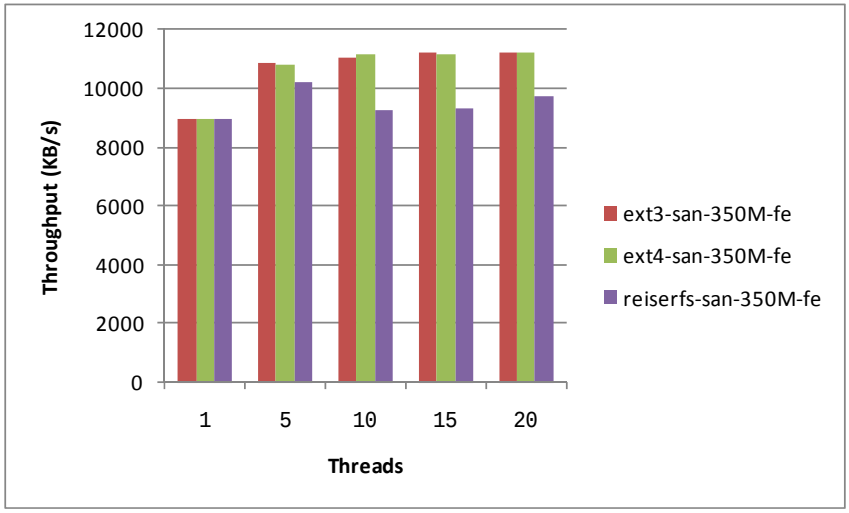
File system	File size	Threads				
		1	5	10	15	20
ext3	1 MB	9113.39	10880.55	10815.49	10797.83	10788.45
	4 MB	9137.59	11320.19	11305.35	11300.38	11298.93
	350 MB	9130.78	11462.78	11212.00	11149.93	10829.34
ext4	1 MB	9454.51	10880.65	10813.98	10795.59	10787.00
	4 MB	9468.99	11320.16	11305.06	11300.92	11298.14
	350 MB	9304.03	9152.29	8111.72	8239.61	7989.65
ReiserFS	1 MB	9398.80	10880.15	10821.37	10791.64	10785.93
	4 MB	9199.74	11319.81	11305.62	11300.88	11298.02
	350 MB	9192.06	10772.57	10773.75	10418.66	10386.74

ตารางที่ 4.4 ผลการวัดประสิทธิภาพการอ่านข้อมูลในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที)

File system	File size	Threads				
		1	5	10	15	20
ext3	1 MB	8958.10	10826.10	11120.04	11218.16	11266.58
	4 MB	8995.63	11264.33	11338.20	11362.96	11375.39
	350 MB	8985.19	10884.70	11059.51	11189.60	11223.47
ext4	1 MB	8958.25	10826.10	11119.42	11218.61	11266.28
	4 MB	8990.42	11406.09	11338.29	11362.83	11375.21
	350 MB	8982.72	10819.44	11166.67	11142.79	11242.10
ReiserFS	1 MB	8938.70	10826.23	11120.77	11218.50	11292.64
	4 MB	8980.69	11407.70	11338.91	11410.40	11375.32
	350 MB	8983.88	10228.15	9243.77	9299.58	9714.55



รูปที่ 4.6 ความเร็วในการเขียนไฟล์ขนาด 350 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที



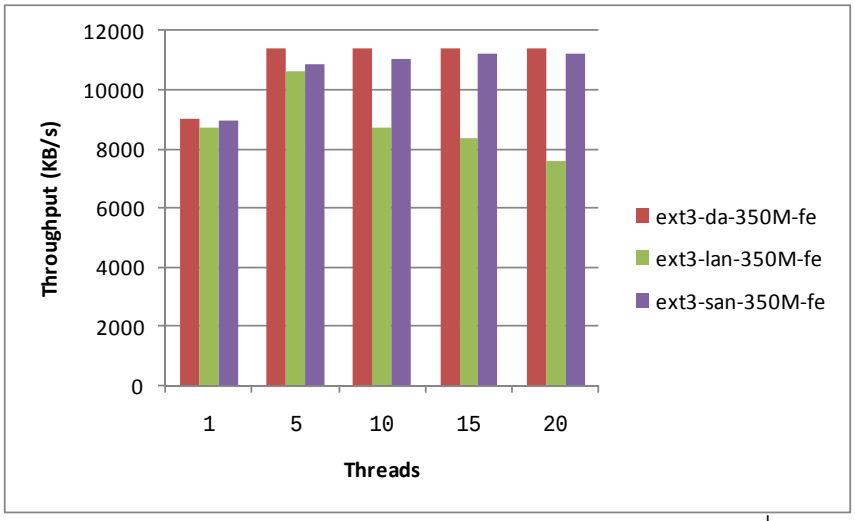
รูปที่ 4.7 ความเร็วในการอ่านไฟล์ขนาด 350 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที

จากผลการทดลองรับ-ส่งข้อมูลบนเครือข่ายที่ความเร็ว 100 เมกกะบิตต่อวินาทีสรุปได้ว่า การเชื่อมต่อแบบรวมส่วนจัดเก็บข้อมูล ระบบไฟล์ ext3 และ ext4 ให้ประสิทธิภาพสูงใกล้เคียงกัน โดยระบบไฟล์ ext3 จะดีในเรื่องการเขียนข้อมูลส่วนการอ่านข้อมูลจะอยู่ในระดับปานกลาง แต่ระบบไฟล์ ext4 จะให้ประสิทธิภาพในการอ่านข้อมูลที่ดีกว่าในแทบทุกการทดสอบ แต่การเขียนข้อมูลขนาดใหญ่ความเร็วจะอยู่ที่ประมาณ 8 เมกกะไบต์ต่อวินาที ส่วนในระบบที่เชื่อมต่อแบบแยกส่วนจัดเก็บข้อมูลระบบไฟล์ ext3 จะให้ประสิทธิภาพสูงสุด และเมื่อนำระบบไฟล์ ext3 มาเปรียบเทียบการเชื่อมต่อต่างๆ ได้แก่ การเชื่อมต่อพื้นที่จัดเก็บบนเครื่องจัดการโดยตรง (ผลการทดสอบ ดัง

ตารางที่ 4.5) การเชื่อมต่อแบบรวมส่วนจัดการ และ การเชื่อมต่อแบบแยกส่วนจัดเก็บข้อมูล พบว่ารูปแบบความเร็วที่ได้ออกมาในลักษณะเดียวกัน คือ ระบบแยกส่วนจัดเก็บข้อมูลให้ประสิทธิภาพดีกว่า ระบบที่รวมส่วนจัดเก็บข้อมูล และใกล้เคียงกับระบบที่ต่อเข้ากับฮาร์ดดิสก์โดยตรงดังรูปที่ 4.8

ตารางที่ 4.5 ความเร็วในการเขียน-อ่านข้อมูลบนระบบไฟล์ ext3 ที่เครื่องจัดการโดยตรงบนเครือข่ายความเร็วระดับ 100 เมกกะบิตต่อวินาที

Operation	File size	Threads				
		1	5	10	15	20
Write	1 MB	9103.83	10875.26	10817.43	10796.59	10784.73
	4 MB	9135.84	11319.91	11304.73	11299.26	11298.33
	350 MB	9424.31	11463.37	11459.40	11464.70	11420.92
Read	1 MB	8939.55	10828.15	11121.86	11218.66	11266.45
	4 MB	8995.30	11264.71	11338.27	11362.74	11375.19
	350 MB	8988.68	11399.73	11399.55	11395.05	11394.11



รูปที่ 4.8 เปรียบเทียบประสิทธิภาพระบบไฟล์ ext3 ระหว่างรูปแบบการเชื่อมต่อต่างๆ

การทดสอบการรับ-ส่งข้อมูลในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาทีผลการทดสอบดังตารางที่ 4.6 และตารางที่ 4.7 ระบบไฟล์ ext4 นั้นให้ประสิทธิภาพในการเขียนข้อมูลสูงที่สุดในทุกการทดสอบตัวอย่างเช่นไฟล์ขนาด 4 เมกกะไบต์ ดังรูปที่ 4.9 และความเร็วในการเขียนข้อมูลจะต่างกันชัดเจนขึ้นเมื่อทดสอบที่ไฟล์ขนาด 350 เมกกะไบต์ ดังรูปที่ 4.10 ส่วนประสิทธิภาพด้านการอ่านข้อมูล ไฟล์ขนาด 1 และ 4 เมกกะไบต์ ระบบไฟล์ทั้ง 3 ให้

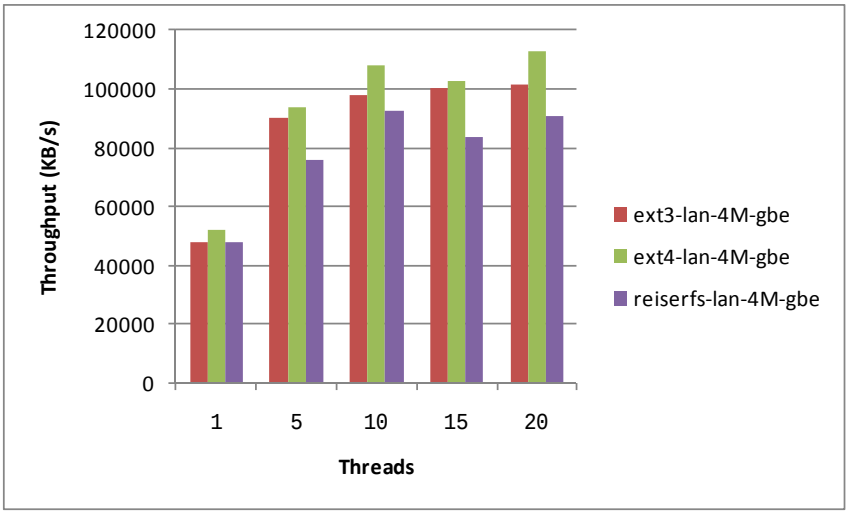
ประสิทธิภาพที่ใกล้เคียงกันดังรูปที่ 4.11 แต่เมื่อทดสอบด้วยไฟล์ขนาด 350 เมกกะไบต์ ระบบไฟล์ ext4 มีประสิทธิภาพสูงสุด รองลงมาเป็นระบบไฟล์ ext3 ซึ่งประสิทธิภาพต่างกันไม่มาก ส่วนระบบไฟล์ ReiserFS ที่มีการใช้งานพร้อมกัน 5 ไฟล์พร้อมกันขึ้นไปความเร็วจะลดลงอยู่ที่ประมาณ 10 เมกกะไบต์ต่อวินาทีเท่านั้นดังรูปที่ 4.12

ตารางที่ 4.6 ผลการวัดประสิทธิภาพการเขียนข้อมูลในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที)

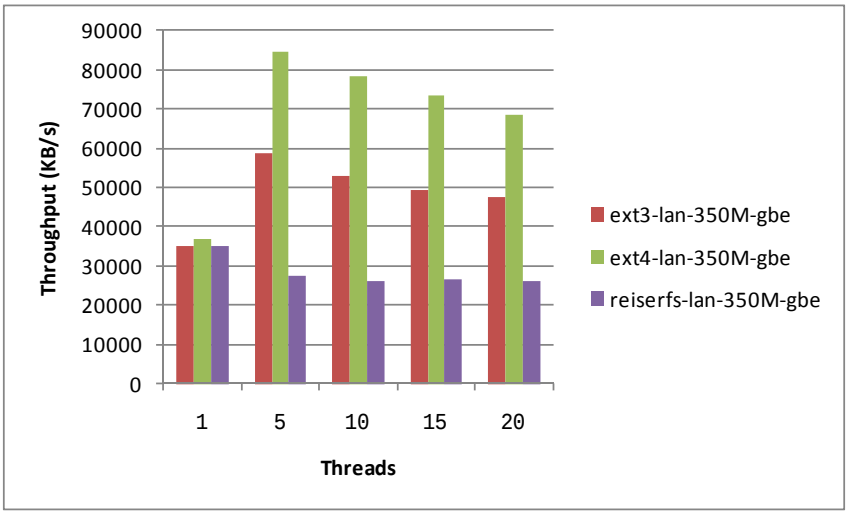
File system	File size	Threads				
		1	5	10	15	20
ext3	1 MB	46192.73	71982.68	91316.06	88115.38	81507.24
	4 MB	47737.86	90276.21	98069.85	100135.52	101291.27
	350 MB	35210.88	58523.27	52877.91	49302.87	47690.49
ext4	1 MB	51613.05	85501.51	106319.35	105754.76	105254.34
	4 MB	52332.27	93541.59	108073.13	102552.91	112629.84
	350 MB	36930.22	84444.56	78123.87	73378.01	68721.69
ReiserFS	1 MB	44504.16	41562.67	27624.23	89482.12	77967.92
	4 MB	47824.83	76005.38	92778.11	83893.94	90617.57
	350 MB	35101.40	27430.00	26076.41	26385.90	25938.94

ตารางที่ 4.7 ผลการวัดประสิทธิภาพการอ่านข้อมูลในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที)

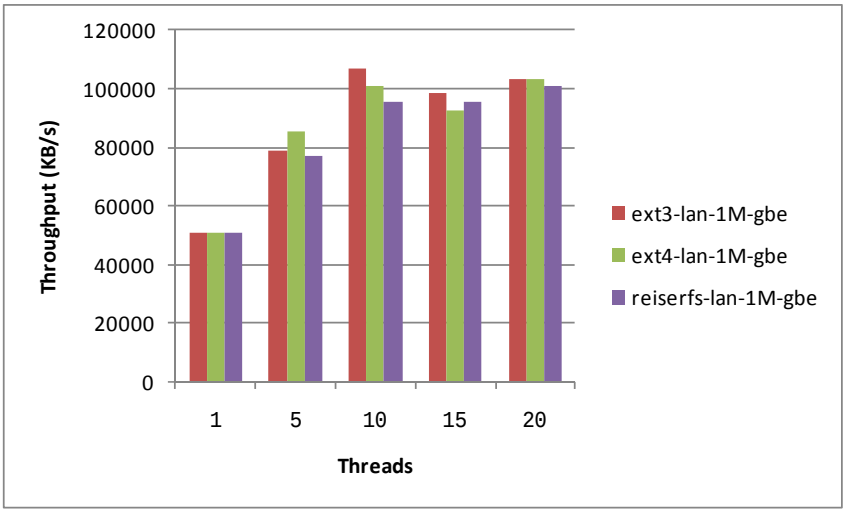
File system	File size	Threads				
		1	5	10	15	20
ext3	1 MB	51095.46	78669.18	107019.04	98399.85	103056.52
	4 MB	51915.18	82133.39	109901.58	111065.68	64941.95
	350 MB	40274.41	29673.88	18781.53	17928.24	17439.74
ext4	1 MB	51092.91	85475.42	100773.63	92259.06	103440.15
	4 MB	52012.66	82323.68	107560.32	110706.32	111234.52
	350 MB	42755.38	30115.94	20113.15	18742.88	18293.94
ReiserFS	1 MB	51077.28	77314.89	95231.33	95795.25	100672.37
	4 MB	52041.77	80708.12	66592.86	110408.24	110374.08
	350 MB	45356.62	13680.86	10274.86	9690.73	9796.14



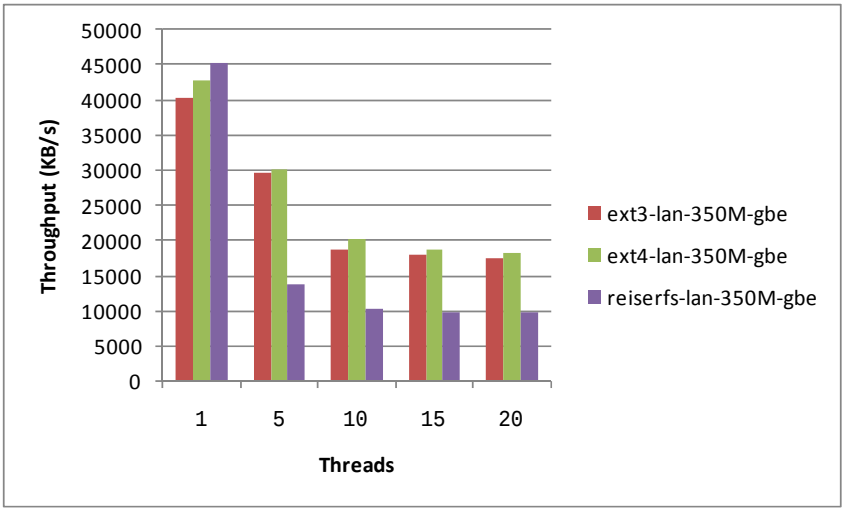
รูปที่ 4.9 ความเร็วในการเขียนข้อมูลขนาด 4 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที



รูปที่ 4.10 ความเร็วในการเขียนข้อมูลขนาด 350 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที



รูปที่ 4.11 ความเร็วในการอ่านข้อมูลขนาด 1 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที



รูปที่ 4.12 ความเร็วในการอ่านข้อมูลขนาด 350 เมกกะไบต์ในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที

ส่วนการทดสอบในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะไบต์ต่อวินาทีผลการทดสอบดังตารางที่ 4.8 และตารางที่ 4.9 ระบบไฟล์ ext4 ให้ประสิทธิภาพในการเขียนข้อมูลสูงที่สุดในทุกการทดสอบ ลักษณะเดียวกับเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ไฟล์ขนาด1 และ 4 เมกกะไบต์จะมีความเร็วในการเขียนข้อมูลต่างกันไม่มากดังรูปที่ 4.13 และต่างกันชัดเจนเมื่อมีการเขียนข้อมูลที่ขนาด 350 เมกกะไบต์ดังรูปที่ 4.14 ส่วนความเร็วในการอ่านข้อมูลระบบไฟล์ทั้ง 3 มีประสิทธิภาพใกล้เคียงกันสำหรับการอ่านไฟล์ขนาด 1 และ 4 เมกกะไบต์ ดังรูปที่ 4.15 แต่ไฟล์

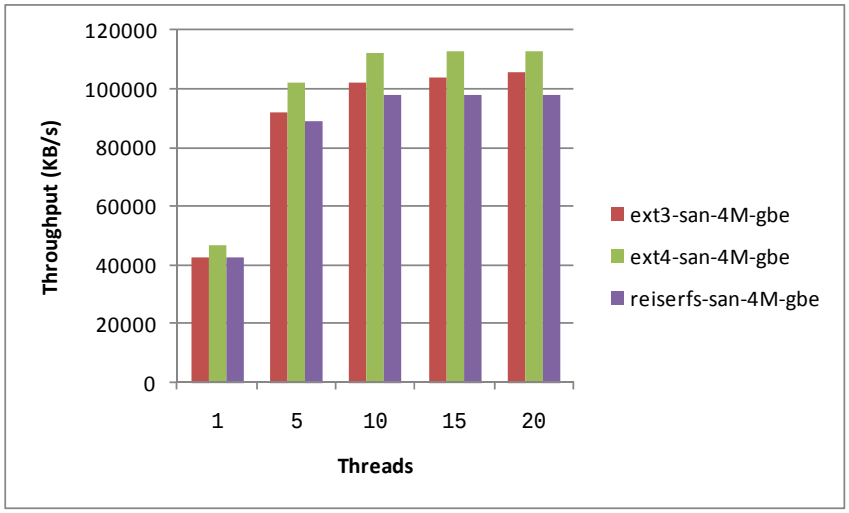
ขนาด 350 เมกกะไบต์ ระบบไฟล์ ext3 และ ext4 มีความเร็วใกล้เคียงกัน และเร็วกว่าระบบไฟล์ ReiserFS ดังรูปที่ 4.16

ตารางที่ 4.8 ผลการวัดประสิทธิภาพการเขียนข้อมูลในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที)

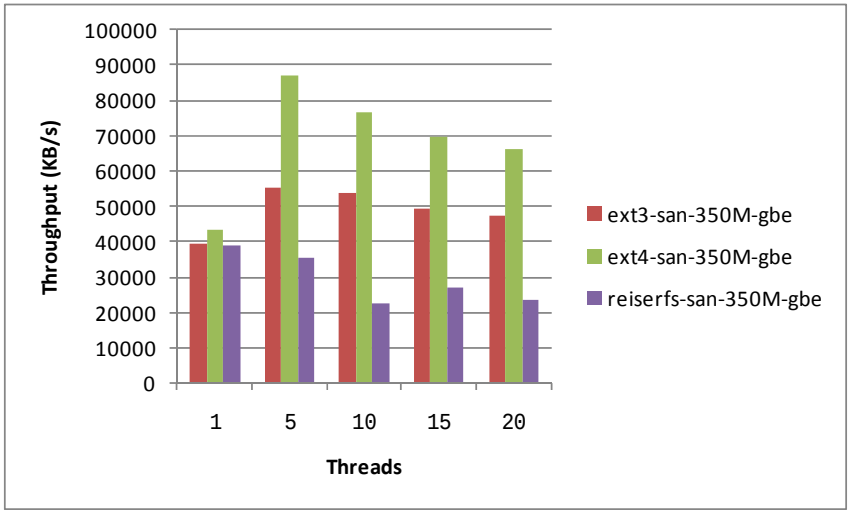
File system	File size	Threads				
		1	5	10	15	20
ext3	1 MB	42148.52	88777.51	97200.65	100252.25	99945.79
	4 MB	42660.41	92082.83	101962.85	103579.90	105453.81
	350 MB	39484.13	55484.84	53875.88	49314.98	47487.61
ext4	1 MB	45156.04	99328.78	106857.15	106791.03	107122.81
	4 MB	46739.83	102304.48	112295.47	112611.40	112689.72
	350 MB	43469.69	87108.02	76797.41	69764.29	66018.75
ReiserFS	1 MB	42514.44	86225.66	95636.54	94478.36	94240.89
	4 MB	42531.09	89033.08	97739.18	98019.84	98048.36
	350 MB	39127.66	35257.42	22798.58	27142.22	23390.51

ตารางที่ 4.9 ผลการวัดประสิทธิภาพการอ่านข้อมูลในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที (หน่วย กิโลไบต์ต่อวินาที)

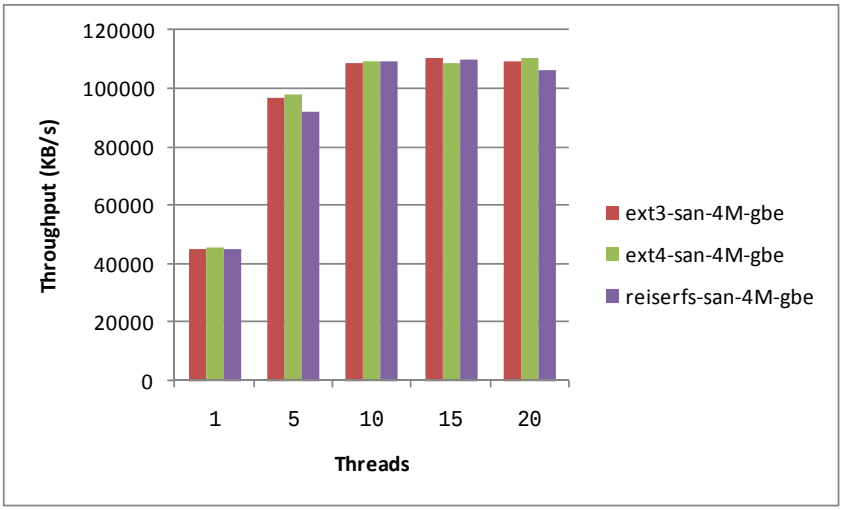
File system	File size	Threads				
		1	5	10	15	20
ext3	1 MB	44833.79	96420.48	105324.11	91294.98	102428.17
	4 MB	45244.22	96442.74	108874.73	110575.53	109432.94
	350 MB	45570.23	28856.71	19471.94	18015.05	17589.80
ext4	1 MB	45122.25	86010.51	101306.74	101991.01	98179.67
	4 MB	45371.41	97981.51	109291.22	108738.11	110117.40
	350 MB	45852.82	24484.15	19160.82	19146.77	17332.26
ReiserFS	1 MB	44067.49	64675.30	103812.34	107520.46	100124.51
	4 MB	45045.62	91984.90	109063.45	109634.96	106005.56
	350 MB	45487.82	14284.24	11021.39	10155.29	10070.95



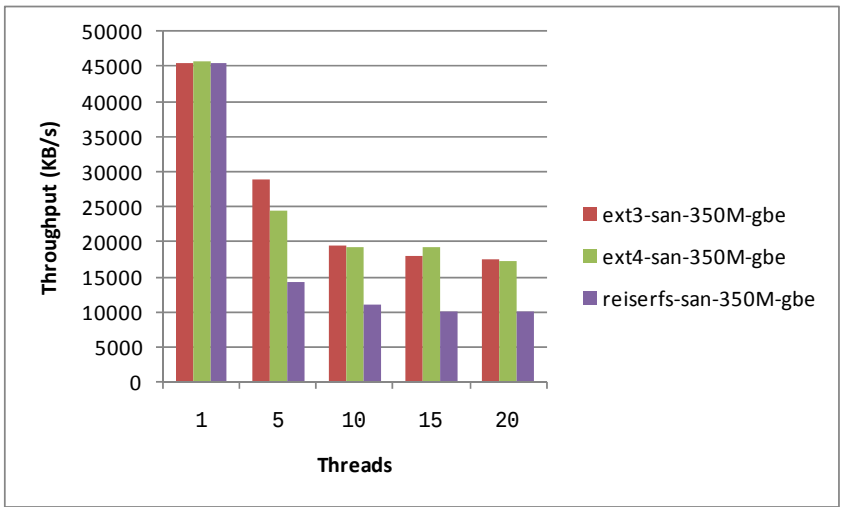
รูปที่ 4.13 ความเร็วในการเขียนข้อมูลขนาด 4 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที



รูปที่ 4.14 ความเร็วในการเขียนข้อมูลขนาด 350 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที



รูปที่ 4.15 ความเร็วในการเขียนข้อมูลขนาด 4 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที

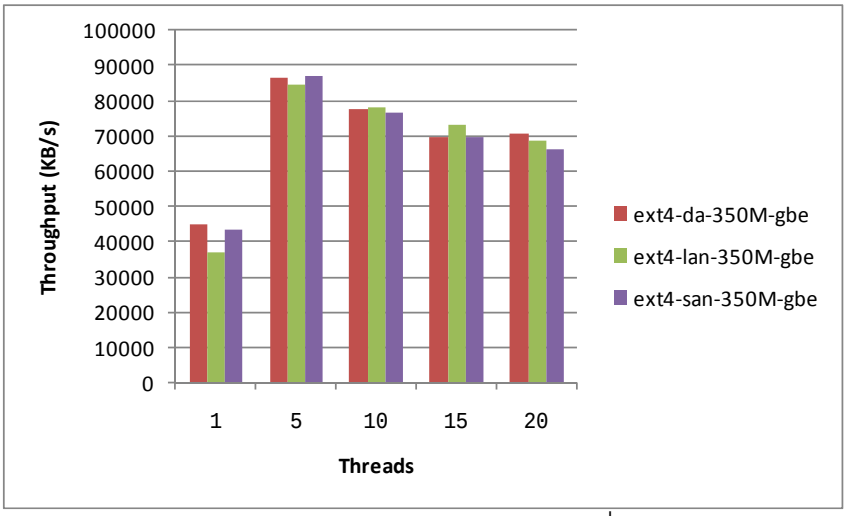


รูปที่ 4.16 ความเร็วในการอ่านข้อมูลขนาด 350 เมกกะไบต์ในเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที

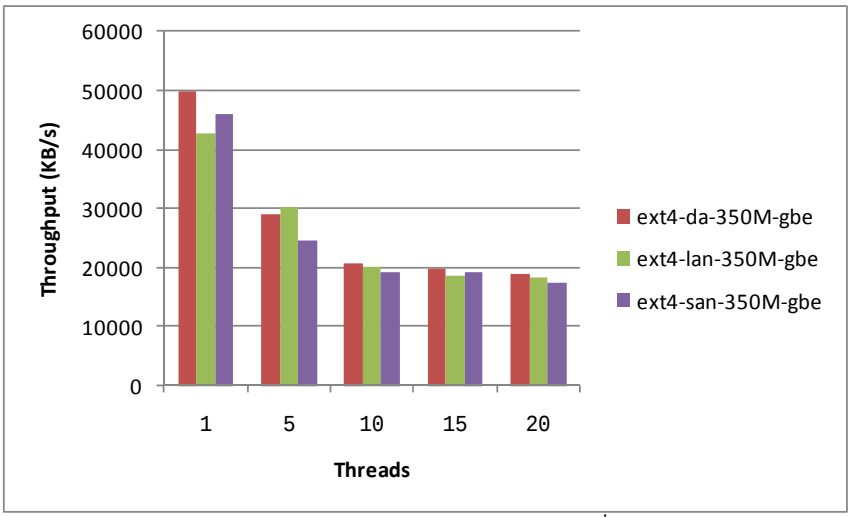
จากผลการทดลองรับ-ส่งข้อมูลบนเครือข่ายความเร็ว 1 กิกะบิตต่อวินาที พบว่าระบบไฟล์ ext4 ให้ประสิทธิภาพดีที่สุดในทุกการทดสอบทั้งการเชื่อมต่อเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลและแยกส่วนจัดเก็บข้อมูล เมื่อนำระบบไฟล์ ext4 มาเปรียบเทียบการเชื่อมต่อต่างๆ ได้แก่ การเชื่อมต่อพื้นที่จัดเก็บบนเครื่องจัดการโดยตรง (ผลการทดลองดังตารางที่ 4.10), การเชื่อมต่อแบบรวมส่วนจัดการ และ การเชื่อมต่อแบบแยกส่วนจัดเก็บข้อมูล พบว่ารูปแบบความเร็วที่ได้ออกมาในลักษณะเดียวกัน คือ การเชื่อมต่อทุกรูปแบบให้ประสิทธิภาพใกล้เคียงกัน ดังรูปที่ 4.17 และรูปที่ 4.18

ตารางที่ 4.10 ความเร็วในการเขียน-อ่านข้อมูลบนระบบไฟล์ ext4 ที่เครื่องจัดการโดยตรงบนเครือข่ายความเร็วระดับ 1 กิกะบิตต่อวินาที

Operation	File size	Threads				
		1	5	10	15	20
Write	1 MB	50983.02	75100.57	96246.99	106743.46	103832.55
	4 MB	52770.59	92628.79	112037.06	105981.59	112728.31
	350 MB	44863.94	86721.79	77581.24	69782.76	70540.46
Read	1 MB	51452.00	77437.78	91979.54	100209.57	98255.03
	4 MB	52081.49	84977.94	108845.16	109157.69	109758.90
	350 MB	49735.40	29086.78	20836.66	19655.37	18896.24

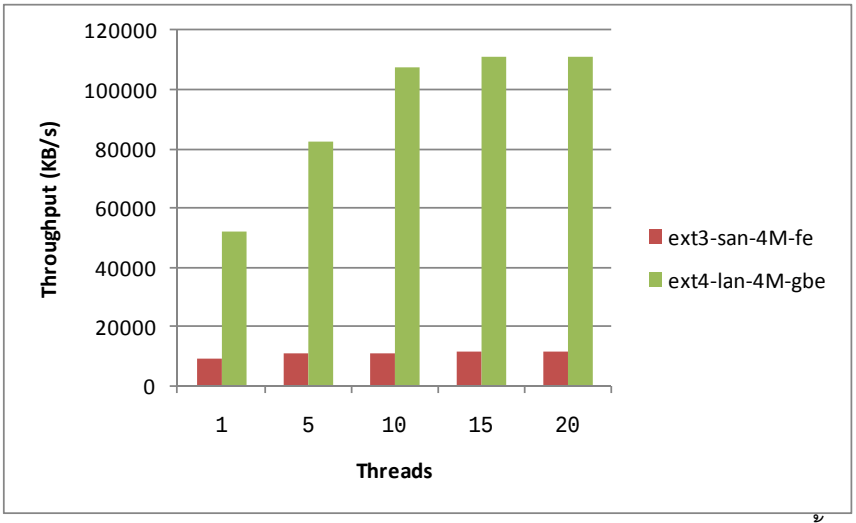


รูปที่ 4.17 เปรียบเทียบความเร็วในการเขียนข้อมูลบนการเชื่อมต่อเครือข่ายแบบต่างๆ

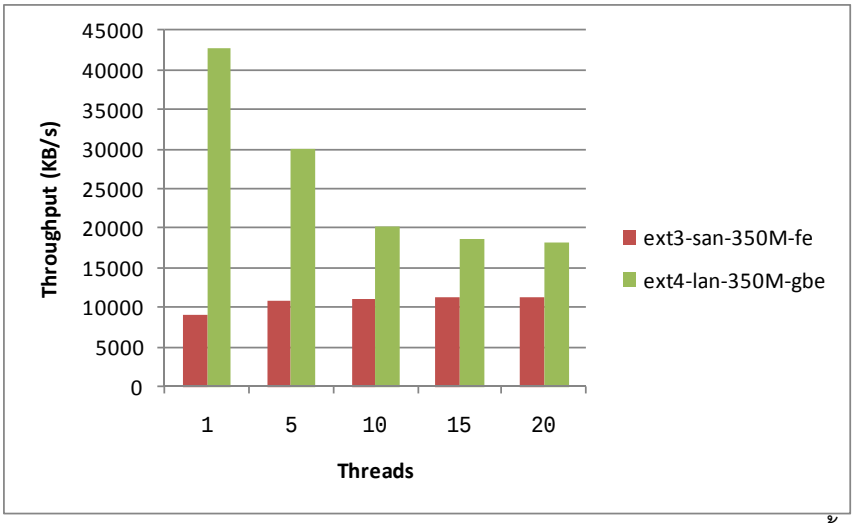


รูปที่ 4.18 เปรียบเทียบความเร็วในการอ่านข้อมูลบนการเชื่อมต่อเครือข่ายแบบต่างๆ

จากการทดสอบทั้งหมดพบว่าในเครือข่ายความเร็วระดับ 100 เมกกะบิตต่อวินาที การตั้งค่าที่เหมาะสมที่สุดคือ การใช้ระบบไฟล์ ext3 และเชื่อมต่อเครือข่ายแบบแยกส่วนจัดเก็บข้อมูล ส่วนเครือข่ายความเร็วระดับ 1 กิกะบิตต่อวินาที การตั้งค่าที่เหมาะสมที่สุดคือ การใช้ระบบไฟล์ ext4 และเชื่อมต่อเครือข่ายแบบรวมส่วนจัดเก็บข้อมูล เมื่อนำระบบที่ตั้งค่าทั้ง 2 แบบมาเปรียบเทียบกัน การเขียน-อ่านข้อมูลที่ขนาด 1 และ 4 เมกกะไบต์ ระบบที่ความเร็วเครือข่ายระดับ 1 กิกะบิตต่อวินาที มีความเร็วมากกว่า ระบบที่ความเร็วเครือข่ายระดับ 100 เมกกะบิต อย่างน้อย 5 เท่า และ 11 เท่า เมื่อมีการใช้งานพร้อมกันหลายไฟล์ ดังรูปที่ 4.19 ส่วนข้อมูลขนาด 350 เมกกะไบต์ การเขียนข้อมูลจะอยู่ในลักษณะเดียวกัน แต่ลดความต่างมาที่ 3 ถึง 7 เท่า ตามลำดับ ส่วนความเร็วการอ่านข้อมูลจะให้ลักษณะที่แตกต่างกันโดยที่การใช้งานเพียงไฟล์เดียวจะให้ประสิทธิภาพสูงที่สุด และความเร็วจะลดลงเรื่อยๆจนถึงประมาณ 18 เมกกะไบต์ต่อวินาทีดังรูปที่ 4.20



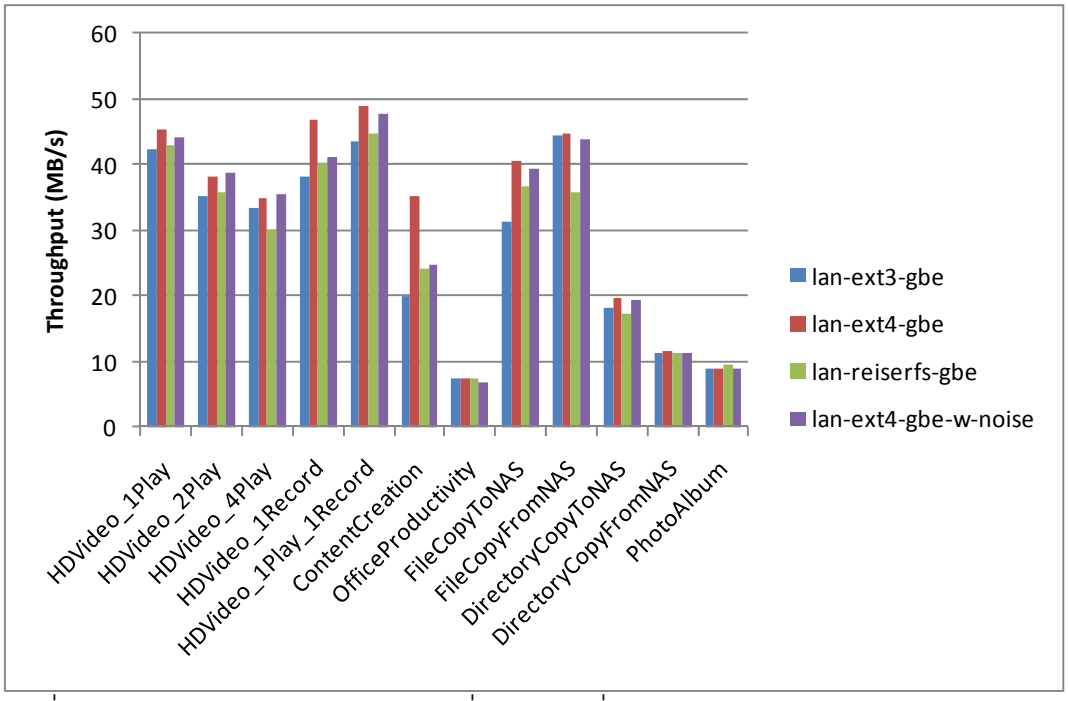
รูปที่ 4.19 เปรียบเทียบความเร็วในการอ่านข้อมูลขนาด 4 เมกกะไบต์ระหว่างการตั้งค่าที่ดีที่สุดของเครือข่ายแต่ละความเร็ว



รูปที่ 4.20 เปรียบเทียบความเร็วในการอ่านข้อมูลขนาด 350 เมกกะไบต์ระหว่างการตั้งค่าที่ดีที่สุดของเครือข่ายแต่ละความเร็ว

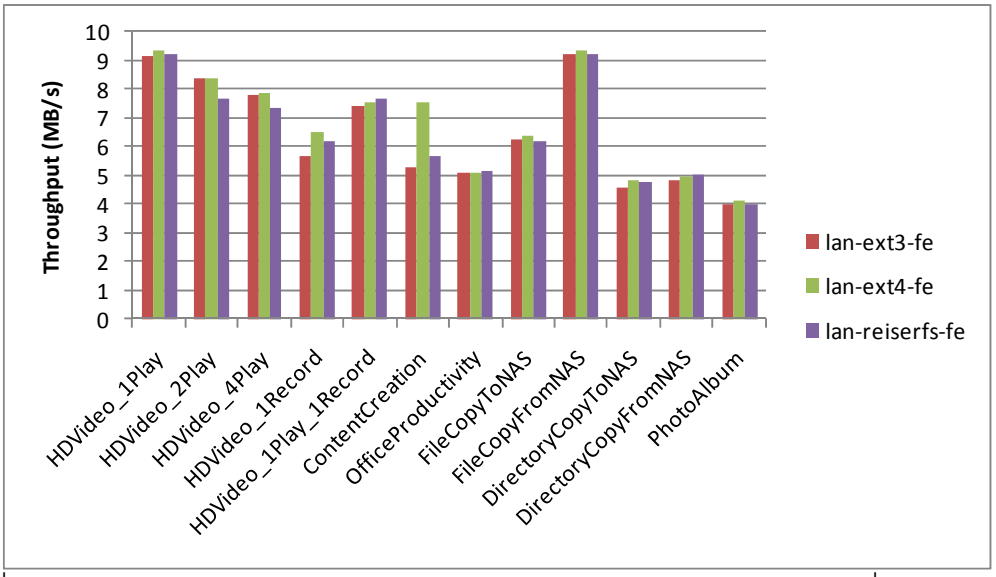
4.1.2 ผลกระทบจากปัจจัยภายนอก

การทดสอบเบื้องต้นจะสนใจการตั้งค่าที่มีประสิทธิภาพสูงที่สุดก่อน ซึ่งในระบบเครือข่ายความเร็ว 1 กิกะบิตต่อวินาที ระบบไฟล์ ext4 ที่การเชื่อมต่อแบบรวมส่วนจัดเก็บข้อมูลมีประสิทธิภาพ และเหมาะสมที่สุด เมื่อทดสอบการรบกวนจากเครื่องลูกข่ายที่ทำการทดสอบพบว่าประสิทธิภาพลดลงเพียงเลยน้อย แต่อย่างไรก็ตามประสิทธิภาพยังคงสูงกว่าระบบไฟล์อื่นที่เชื่อมต่อแบบเดียวกันที่ไม่มีการรบกวน ดังรูปที่ 4.21

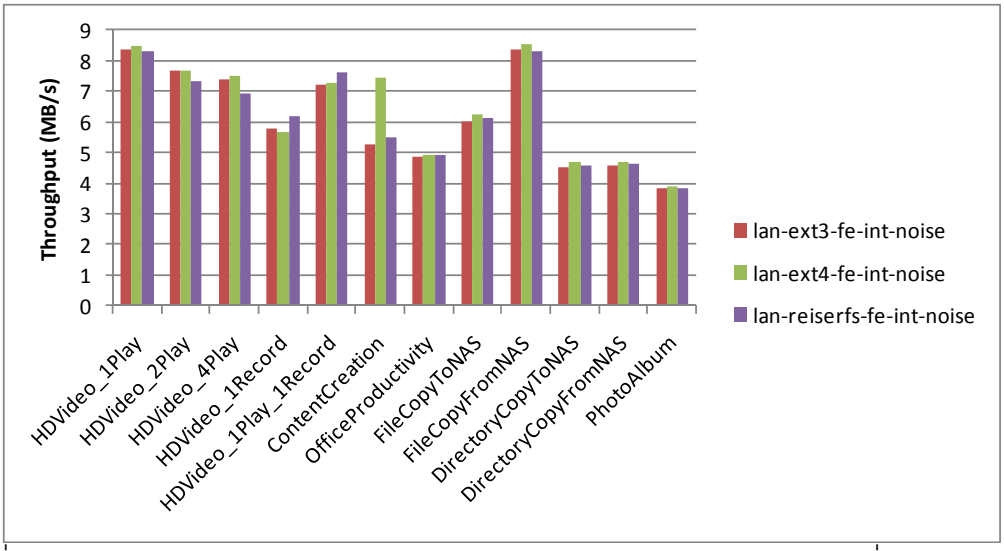


รูปที่ 4.21 การทดสอบการรบกวนจากเครื่องลูกข่ายที่ทำการทดสอบในเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 1 กิกะบิตต่อวินาที

ส่วนเครือข่ายที่ความเร็วระดับ 100 เมกกะบิตต่อวินาทีการเชื่อมต่อแบบรวมส่วนจัดเก็บข้อมูลระบบไฟล์ ext4 ให้ผลการทดสอบที่ดีที่สุดเกือบทุกประเภทการใช้งานดังรูปที่ 4.22 และเมื่อทดสอบการรบกวนจากเครื่องลูกข่ายพบว่าระบบไฟล์ ext4 ยังคงให้ประสิทธิภาพสูงที่สุดดังรูปที่ 4.23 เมื่อพิจารณาเปรียบเทียบระบบที่มีและไม่มีการรบกวน ระบบที่มีการรบกวนประสิทธิภาพจะลดลงไปประมาณ 1 เมกกะไบต์ต่อวินาที หรือเท่ากับ 10 เมกกะบิตต่อวินาที ตามความเร็วการดาวร์โหลดข้อมูลจากอินเทอร์เน็ต

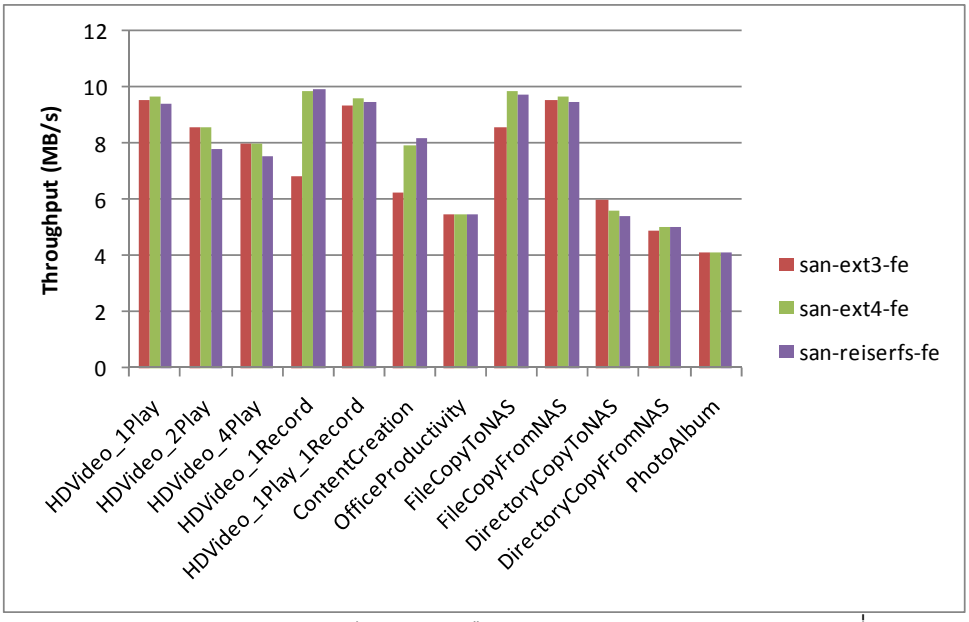


รูปที่ 4.22 การวัดประสิทธิภาพระบบไฟล์บนเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที ด้วยโปรแกรม Intel NASPT

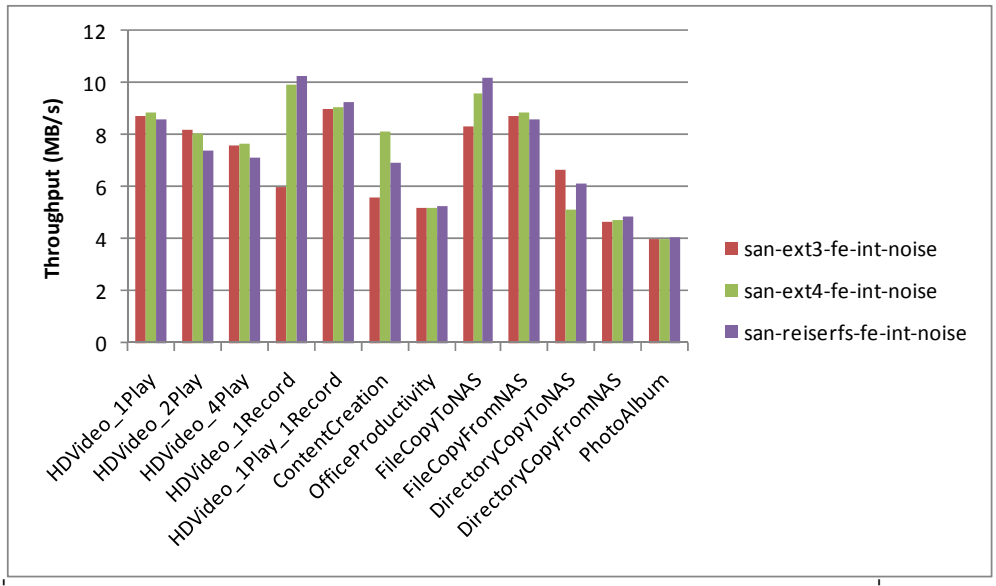


รูปที่ 4.23 การวัดประสิทธิภาพระบบไฟล์บนเครือข่ายแบบรวมส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที โดยมีการรบกวนจากเครื่องลูกข่ายด้วยโปรแกรม Intel NASPT

ส่วนการเชื่อมต่อแบบแยกส่วนจัดเก็บข้อมูล ระบบไฟล์ ext4 ให้ประสิทธิภาพสูงที่สุดเกือบทุกการทดสอบดังรูปที่ 4.24 และเมื่อทดสอบการรบกวนจากเครื่องลูกข่าย ระบบไฟล์ ext4 ยังคงให้ประสิทธิภาพสูงที่สุดในหลายๆการทดสอบ และได้รับผลกระทบเพียงบางการทดสอบเท่านั้น



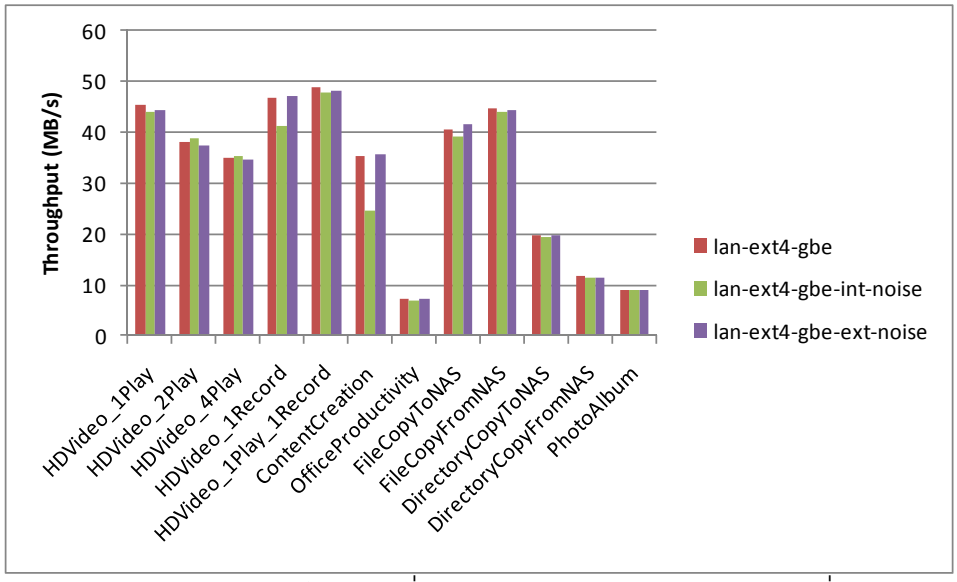
รูปที่ 4.24 การวัดประสิทธิภาพระบบไฟล์บนเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที ด้วยโปรแกรม Intel NASPT



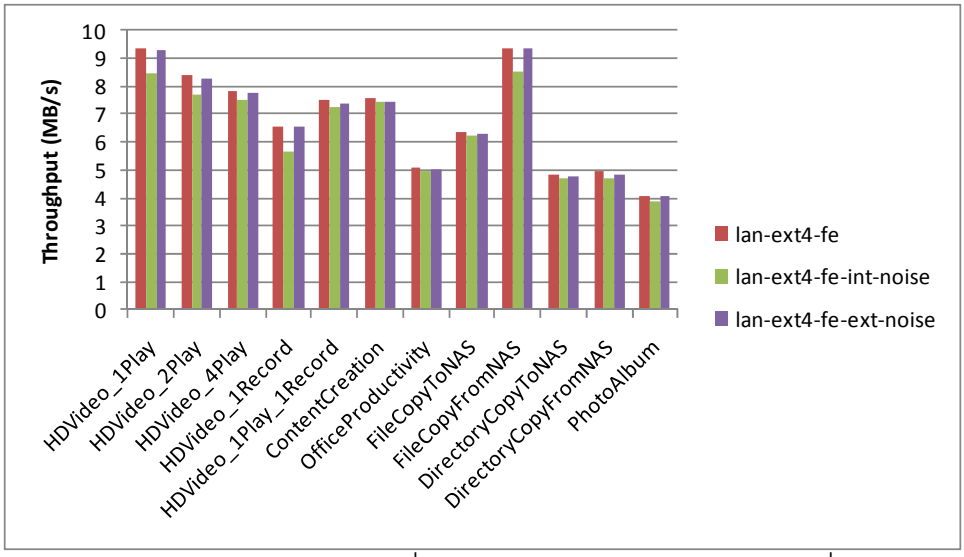
รูปที่ 4.25 การวัดประสิทธิภาพระบบไฟล์บนเครือข่ายแบบแยกส่วนจัดเก็บข้อมูลที่ความเร็ว 100 เมกกะบิตต่อวินาที โดยมีการรบกวนจากเครื่องลูกข่ายด้วยโปรแกรม Intel NASPT

การทดสอบการรบกวนจากเครื่องอื่นภายในระบบพบว่าประสิทธิภาพการใช้งานระบบจัดเก็บข้อมูลนั้นไม่ได้รับผลกระทบจากการรบกวนภายนอกดังรูปที่ 4.26 และรูปที่ 4.27 ซึ่งเป็นเพราะว่าสวิตช์ที่ใช้ทำการทดสอบมีเทคโนโลยีการเรียนรู้ MAC Address ทำให้การส่งข้อมูลจะส่ง

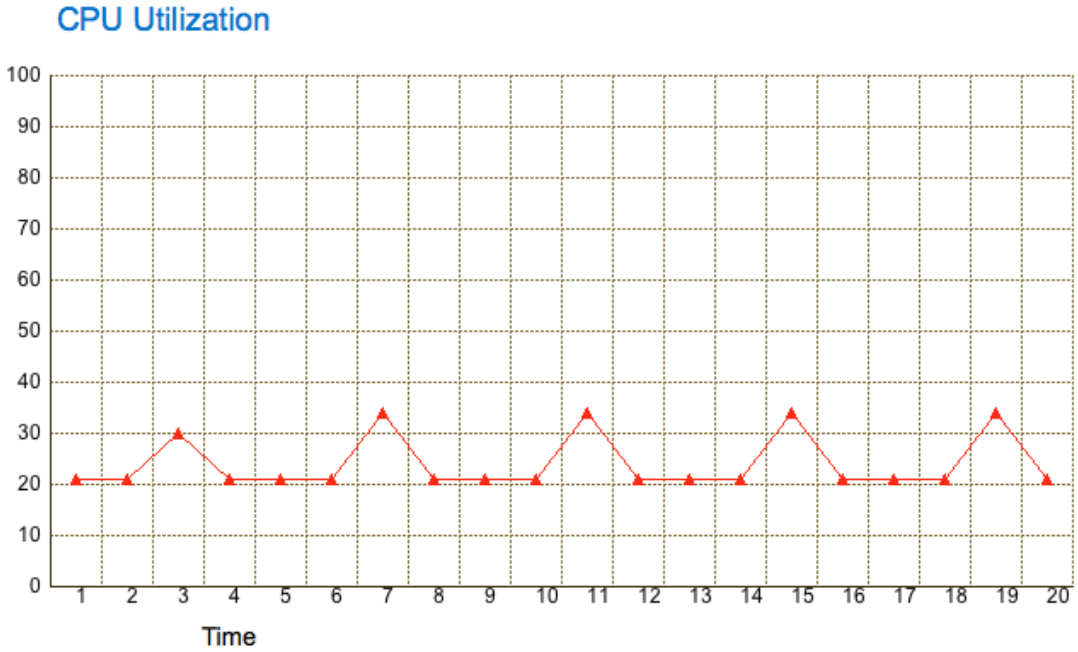
ต่อจากพอร์ทต้นทางไปยังพอร์ทปลายทางที่เชื่อมต่ออินเทอร์เน็ตเท่านั้น และเมื่อตรวจสอบการใช้งานซีพียูของสวิตช์พบว่าใช้ไปเพียง 20-35 เปอร์เซ็นต์เท่านั้นดังรูปที่ 4.28



รูปที่ 4.26 เปรียบเทียบระบบบันทึกข้อมูลที่ไม่มีการรบกวน มีการรบกวนจากเครื่องลูกข่าย และมีการรบกวนจากเครื่องอื่นภายในระบบ ของเครื่องข่ายความเร็วระดับ 1 กิกะบิตต่อวินาที



รูปที่ 4.27 เปรียบเทียบระบบบันทึกข้อมูลที่ไม่มีการรบกวน มีการรบกวนจากเครื่องลูกข่าย และมีการรบกวนจากเครื่องอื่นภายในระบบ ของเครื่องข่ายความเร็วระดับ 100 เมกกะบิตต่อวินาที



รูปที่ 4.28 การใช้ซีพียูของสวิตช์ขณะทำการทดสอบการรบกวนของระบบ

การทดสอบในส่วนนี้ทำให้ทราบเพิ่มเติมว่าในการใช้งานจริงแบบผู้ใช้คนเดียว ระบบไฟล์ ext4 ให้ประสิทธิภาพที่ดีในการใช้งานหลายๆประเภท รวมไปถึงการใช้งานในเครือข่ายความเร็วระดับ 100 เมกกะบิตต่อวินาที ที่การเชื่อมต่อเครือข่ายทั้ง 2 รูปแบบ นอกจากนี้ยังพบว่าปัจจัยที่สำคัญต่อประสิทธิภาพคือแบนด์วิดท์ในการเชื่อมต่อส่วนต่างๆในเครือข่าย

4.2 การทดสอบการทำงานของแบบจำลองที่ออกแบบ

จากการทดสอบข้างต้นทำให้ทราบว่าโพรโทคอลที่ใช้ในการเชื่อมต่อฮาร์ดดิสก์ที่มีประสิทธิภาพสูงที่สุดคือ โพรโทคอล iSCSI ดังนั้นระบบค้นหาตามแบบจำลองที่นำเสนอ จะทำการค้นหาบริการฮาร์ดดิสก์ที่ใช้แต่โพรโทคอล iSCSI เครื่องให้บริการฮาร์ดดิสก์จะติดตั้งโปรแกรม Avahi ที่ทำหน้าที่ตอบกลับรายละเอียดที่ให้บริการ โดยมีการตั้งค่าดังรูปที่ 4.29 ซึ่งในส่วนของข้อมูล <type>_iscsi._tcp</type> จะเป็นการประกาศชนิดของบริการว่าเป็นฮาร์ดดิสก์ iSCSI และข้อมูล <txt-record>Target=…</txt-record> จะเป็นชื่อของฮาร์ดดิสก์ที่ให้บริการอยู่ ซึ่งแต่ละเครื่องจะมีข้อมูลไม่ซ้ำกัน

```
<?xml version="1.0" standalone='no'?><!--*-nxml-*-->
<!DOCTYPE service-group SYSTEM "avahi-service.dtd">
<service-group>
<name replace-wildcards="yes">%h</name>
<service>
<type>_iscsi._tcp</type>
<port>3260</port>
<txt-record>Target=iqn.2011-09.com.isel:node1.disk1</txt-record>
</service>
</service-group>
```

รูปที่ 4.29 โครงสร้างไฟล์ที่ใช้ในการประกาศบริการฮาร์ดดิสก์

ส่วนเครื่องจัดการจะติดตั้งโปรแกรม Avahi เช่นกัน แต่ทำหน้าที่ในการประกาศหาเครื่องที่ให้บริการฮาร์ดดิสก์ที่อยู่ในเครือข่าย การทำงานทั้งหมดจะอยู่ในรูปของสคริปต์คำสั่งที่ทำงานตามช่วงเวลา เมื่อให้สั่งทำงานขั้นแรกจะทำการเรียกโปรแกรม Avahi เพื่อค้นหาฮาร์ดดิสก์ในเครือข่าย เมื่อได้รับข้อความตอบกลับดังรูปที่ 4.30 และรูปที่ 4.31 ในกรณีมีเครื่องที่ให้บริการฮาร์ดดิสก์ 2 เครื่อง เครื่องจัดการจะได้รับชื่อเครื่อง หมายเลขไอพี พอร์ทที่ให้บริการ และชื่อฮาร์ดดิสก์ที่ใช้ในการเชื่อมต่อ ถ้ามีเครื่องใหม่เพิ่มเข้ามาในระบบมากกว่าครั้งละ 1 เครื่องจะทำการเชื่อมต่อทีละเครื่องจนเสร็จขั้นตอนแล้วถึงจะเชื่อมต่อเครื่องถัดไป เมื่อทำการเชื่อมต่อเรียบร้อยจะเพิ่มชื่อลงไปในฐานข้อมูลแล้วจะตรวจสอบขนาดของฮาร์ดดิสก์ที่พึ่งเชื่อมต่อ แล้วทำการแบ่งพาร์ทิชันด้วยโปรแกรม parted [22] และเพิ่มเข้าสู่ Volume Group จากนั้นจะทำการพื้นที่เสมือนและระบบไฟล์

```
+   eth0 IPv4 storage1                                      _iscsi._tcp          local
=   eth0 IPv4 storage1                                      _iscsi._tcp          local
   hostname = [storage1.local]
   address = [10.0.0.1]
   port = [3260]
   txt = ["Target=iqn.2011-09.com.isel:node1.disk1"]
```

รูปที่ 4.30 ข้อความตอบกลับจากเครื่องที่ให้บริการฮาร์ดดิสก์

```
+   eth0 IPv4 storage1                                      _iscsi._tcp          local
+   eth0 IPv4 storage2                                      _iscsi._tcp          local
=   eth0 IPv4 storage1                                      _iscsi._tcp          local
   hostname = [storage1.local]
   address = [10.0.0.1]
   port = [3260]
   txt = ["Target=iqn.2011-09.com.isel:node1.disk1"]
=   eth0 IPv4 storage2                                      _iscsi._tcp          local
   hostname = [storage2.local]
   address = [10.0.0.2]
   port = [3260]
   txt = ["Target=iqn.2011-09.com.isel:node2.disk1"]
```

รูปที่ 4.31 ข้อความตอบกลับจากเครื่องที่ให้บริการฮาร์ดดิสก์ในกรณีที่มี 2 เครื่อง

การทดสอบทั้งหมดนั้นใช้วิธีจำลองด้วยเครื่องเสมือนผ่านโปรแกรม VMWare โดยเครื่องที่ให้บริการฮาร์ดดิสก์มีฮาร์ดดิสก์แยกต่างหากสำหรับติดตั้งระบบปฏิบัติการ และสำหรับให้บริการโดยฮาร์ดดิสก์สำหรับให้บริการมีขนาดอยู่ที่ 8 กิกะไบต์ การทดลองเริ่มต้นจากเครื่องจัดการเพียงเครื่องเดียว จากนั้นค่อยๆ เพิ่มเครื่องที่ให้บริการฮาร์ดดิสก์เข้าไปในระบบ เริ่มจาก 1 เครื่อง เครื่องจัดการสามารถตรวจพบแล้วเชื่อมต่อได้อย่างถูกต้อง รวมไปถึงการขยายพื้นที่จัดเก็บข้อมูล จากนั้นผู้วิจัยได้ทดลองเพิ่มเครื่องที่ให้บริการฮาร์ดดิสก์เข้าไปอีก 3 เครื่อง เครื่องจัดการสามารถตรวจพบเครื่องใหม่ได้ทั้ง 3 เครื่อง และทำการเชื่อมต่อทีละเครื่องๆ จนครบทั้ง 3 เครื่อง ทำให้มีพื้นที่จัดเก็บข้อมูลรวมทั้งหมด 32 กิกะไบต์ และเมื่อทดสองสุ่มเพิ่มจำนวนเครื่องที่ให้บริการฮาร์ดดิสก์ไปเรื่อยๆจนครบ 10 เครื่อง เครื่องจัดการสามารถทำงานได้อย่างถูกต้อง และมีพื้นที่จัดเก็บข้อมูลรวมเท่ากับ 80 กิกะไบต์ ซึ่งตรงกับขนาดของฮาร์ดดิสก์ในระบบรวมกัน

บทที่ 5
บทสรุป

5.1 สรุปผลการวิจัย

งานวิจัยนี้ได้นำเสนอแบบจำลองการสร้างระบบจัดเก็บข้อมูลบนเครือข่ายแบบขยายพื้นที่จัดเก็บได้อัตโนมัติ โดยเลือกใช้ส่วนประกอบจากเทคโนโลยีสำเร็จรูปที่มีอยู่มากมายในท้องตลาด และทำการคัดเลือกส่วนประกอบที่ให้ประสิทธิภาพสูงสุดในแต่ละลักษณะงาน ซึ่งทำให้ทราบถึงประสิทธิภาพและผลกระทบของส่วนประกอบทั้งด้านฮาร์ดแวร์ และซอฟต์แวร์

จากการทดสอบพบว่าที่เครือข่ายระดับความเร็ว 100 เมกกะบิตต่อวินาที ระบบไฟล์ ext3 และ ext4 ให้ประสิทธิภาพสูง และเหมาะสมที่สุดในการใช้งานโดยทั่วไป ซึ่งระบบไฟล์ ext3 จะเด่นที่การเขียนข้อมูลพร้อมกัน แต่ระบบไฟล์ ext4 จะเด่นในเรื่องของการอ่านข้อมูล ส่วนการเชื่อมต่อแบบแยกส่วนจัดเก็บข้อมูลจะช่วยเพิ่มประสิทธิภาพในการอ่านข้อมูลพร้อมกันหลายไฟล์มากกว่าระบบที่รวมส่วนจัดเก็บข้อมูลประมาณ 20-30 เปอร์เซ็นต์ และใกล้เคียงกับระบบบันทึกข้อมูลบนเครือข่ายแบบปกติ ส่วนเครือข่ายที่ความเร็วในระดับ 1 กิกะบิตต่อวินาที ระบบไฟล์ ext4 มีประสิทธิภาพสูงที่สุดในทุกการทดสอบ และความเร็วที่ได้สำหรับการเชื่อมต่อรูปแบบต่างๆ ค่อนข้างใกล้เคียงกันรวมถึงระบบบันทึกข้อมูลบนเครือข่ายแบบปกติ ดังนั้นการเชื่อมต่อแบบรวมส่วนจัดเก็บข้อมูลจะเหมาะสมที่สุดเนื่องจากประหยัดค่าใช้จ่ายในส่วนของสวิตช์ และการ์ดเชื่อมต่อเครือข่าย กรณีถ้าใช้เครือข่ายความเร็วระดับ 100 เมกกะบิตต่อวินาทีอยู่ก่อน และต้องการอัพเกรดระบบเครือข่ายให้เป็นระดับ 1 กิกะบิตต่อวินาที ก็สามารถทำได้อย่างสะดวกเนื่องจากระบบไฟล์ ext3 สามารถแปลงไปเป็นระบบไฟล์ ext4 ได้โดยที่ข้อมูลไม่ศูนย์หาย

ระบบที่ออกแบบและพัฒนาขึ้นมานั้นรองรับการใช้งานทั่วไปได้ดี สามารถขยายพื้นที่จัดเก็บข้อมูลได้เรื่อยๆ เพียงเพิ่มเครื่องบริการฮาร์ดดิสก์เข้าไปในระบบโดยที่ไม่ต้องทำการตั้งค่าใดๆ นอกจากนี้ยังให้ประสิทธิภาพใกล้เคียงกับระบบทั่วไป และมีต้นทุนต่ำเนื่องจากใช้อุปกรณ์สำหรับผู้ใช้ตามบ้านที่หาซื้อได้ทั่วไป

เมื่อพิจารณาต้นทุนในการติดตั้งระบบใหม่พบว่าคอมพิวเตอร์ในท้องตลาดขณะที่ทำการวิจัยพอร์ทอีเทอเน็ตจะอยู่ในระดับความเร็ว 1 กิกะบิตต่อวินาทีทุกรุ่นอยู่แล้ว ดังนั้นต้นทุนที่ต่างกัน

ระหว่างการตั้งค่าต่างๆ จะอยู่ที่ สวิตช์ การ์ดเชื่อมต่อเครือข่ายเพิ่มเติม และสายสำหรับเชื่อมต่อเครือข่ายเท่านั้น

5.2 ปัญหาและข้อจำกัดที่พบจากการวิจัย

5.2.1 การเชื่อมต่อเครือข่ายที่ระดับความเร็ว 1 กิกะบิตต่อวินาที การ์ดเชื่อมต่อเครือข่ายในระดับ 1 กิกะบิตต่อวินาที จะต้องเชื่อมต่อกับคอมพิวเตอร์ที่ระบบบัส PCI-Express, PCI-X หรือระบบบัสแบบอื่นที่มีแบนด์วิดท์มากกว่า 1 กิกะบิตต่อวินาที เนื่องจากระบบบัส PCI ที่ใช้กันทั่วไปมีแบนด์วิดท์เพียงแค่ 533 เมกกะไบต์ต่อวินาที อย่างไรก็ตามการ์ดเชื่อมต่อเครือข่ายที่ใช้ในงานวิจัยทั้งหมดเชื่อมต่อกับระบบบัส PCI-Express

5.2.2 ซอฟต์แวร์ iSCSI Enterprise Target ที่ใช้ในเครื่องที่ให้บริการฮาร์ดดิสก์ การติดตั้งจากไบนารี่แพคเกจจะให้ประสิทธิภาพไม่เท่ากับที่คอมไพล์เองจากซอร์สโค้ด ถึงแม้จะเป็นรุ่นเดียวกัน งานวิจัยนี้จึงทำการคอมไพล์ซอฟต์แวร์จากซอร์สโค้ดแทนการติดตั้งจากไบนารี่แพคเกจของระบบปฏิบัติการ

5.2.3 ระบบไฟล์ ZFS แบบ Native สำหรับระบบปฏิบัติการลินุกซ์ จำเป็นจะต้องใช้ระบบปฏิบัติการ 64 บิต จึงใช้งานได้ ดังนั้นเครื่องจัดการที่ใช้ในงานวิจัยจึงจำเป็นต้องใช้ระบบปฏิบัติการ 64 บิต ส่วนเครื่องอื่นในระบบจะใช้ระบบปฏิบัติการ 32 บิต

5.2.4 การทดสอบประสิทธิภาพทำโดยการรับ-ส่งข้อมูลเป็นชุดต่อเนื่องกัน และโปรแกรมทดสอบที่ทำงานบนเครื่องลูกข่ายใช้ทรัพยากรในการสร้างข้อมูลชุดทดสอบอย่างเต็มที่ ทำให้การตรวจวัดการใช้ทรัพยากรในส่วนอื่นของระบบไม่สามารถทำได้

5.3 ข้อเสนอแนะ

งานวิจัยนี้สามารถพัฒนาเพิ่มเติมได้โดยการเพิ่มความสามารถในการตรวจสอบฮาร์ดดิสก์ที่กำลังจะเสีย และทำการแจ้งเตือนผู้ใช้ จากนั้นจะยุติการเชื่อมต่อฮาร์ดดิสก์นั้น ซึ่งจะช่วยให้ข้อมูลมีความปลอดภัยมากขึ้น

รายการอ้างอิง

[1] Phromchana, V., Nupairoj, N., and Piromsopa, K. Building a Zero-Configuration Extensible Network Storage Based on Open Source Technology. Proceedings of National Computer Science and Engineering Conference, 2008. NCSEC 2008. 12th National Computer Science and Engineering Conference on, p. 599 pp. 20-21 Nov. 2008, 2008.

[2] Phromchana, V., Nupairoj, N., and Piromsopa, K. Performance evaluation of ZFS and LVM (with ext4) for scalable storage system. Proceedings of Computer Science and Software Engineering (JCSSE), 2011 Eighth International Joint Conference on, pp. 250-253. 11-13 May 2011, 2011.

[3] Zero Configuration Networking (Zeroconf) [Online]. Available from: http://www.zeroconf.org/ [2011, August 31].

[4] The Internet Society. RFC 3720 - Internet Small Computer Systems Interface (iSCSI) [Online]. 2004. Available from: http://www.ietf.org/rfc/rfc3720.txt [2011, August 31].

[5] Shuang-Yi, T., Ying-Ping, L., and Du, D. H. C. Performance study of software-based iSCSI security. Security in Storage Workshop, 2002. Proceedings. First International IEEE (2002): 70-79.

[6] The Brantley Coile Company, Inc. ATA over Ethernet. 2009. Available from: http://support.coraid.com/documents/AoEr11.pdf [2011, August 31].

[7] The Internet Society. RFC 3721 - iSCSI Naming and Discovery Standard [Online]. 2004. Available from: http://www.ietf.org/rfc/rfc3721.txt [2011, August 31].

[8] Brinkmann, A., Effert, S., Heidebuer, M., and Vodisek, M. Influence of adaptive data layouts on performance in dynamically changing storage environments. Proceedings of Parallel, Distributed, and Network-Based Processing, 2006. PDP 2006. 14th Euromicro International Conference on, p. 8 pp. 15-17 Feb. 2006, 2006.

[9] EXT3, Journaling Filesystem. 2000. Available from: http://olstrans.sourceforge.net/release/OLS2000-ext3/OLS2000-ext3.html [2011, August 31].

[10] Mathur, A. a. C., Mingming and Bhattacharya, Suparna and Dilger, Andreas and Tomas, Alex and Vivier, Laurent. The new ext4 filesystem: current status and future plans. Linux Symposium (2007).

[11] Microsoft Corporation. NTFS Technical Reference. 2003. Available from: http://technet.microsoft.com/en-us/library/cc778410%28WS.10%29.aspx [2011, August 31].

[12] Reiser, H. T. Enhancing ReiserFS security in linux. Proceedings of DARPA Information Survivability Conference and Exposition, 2003. Proceedings, p. 188 vol.2. 22-24 April 2003, 2003.

[13] Pugh, C., Henderson, P., Silber, K., Carroll, T., and Ying, K. Utilizing zfs for the storage of acquired data. Proceedings of Fusion Engineering, 2009. SOFE 2009. 23rd IEEE/NPSS Symposium on, pp. 1-4. 1-5 June 2009, 2009.

[14] Yasuda, Y., Kawarnoto, S., Ebata, A., Okitsu, J., and Higuchi, T. Concept and evaluation of X-NAS: a highly scalable NAS system. Proceedings of Mass Storage Systems and Technologies, 2003. (MSST 2003). Proceedings. 20th IEEE/11th NASA Goddard Conference on, pp. 219-227. 7-10 April 2003, 2003.

[15] Qingguo, Z., Guanghui, C., Rui, Z., Lian, L., and Bin, H. A Case Study of OSS/COTS-Based System Development: Design and Implementation of an Open Scalable Network Storage Attached System. Proceedings of Pervasive Computing and Applications, 2008. ICPCA 2008. Third International Conference on, pp. 122-127. 6-8 Oct. 2008, 2008.

[16] Yingping, L. and Du, D. H. C. Performance study of iSCSI-based storage subsystems. Communications Magazine, IEEE 41 (2003): 76-82.

[17] Avahi [Online]. Available from: http://avahi.org [2011, August 31].

[18] Microsoft Corporation. Microsoft SMB Protocol and CIFS Protocol Overview. 2011. Available from: http://msdn.microsoft.com/en-us/library/aa365233(VS.85).aspx [2011, September 15].

[19] IOzone Filesystem Benchmark [Online]. 2006. Available from: http://www.iozone.org/ [2011, August 31].

[20] Intel Corporation. Intel(R) NAS Performance Toolkit [Online]. 2009. Available from: http://software.intel.com/en-us/articles/intel-nas-performance-toolkit/ [2011, August 31].

[21] VMware, Inc. VMware Workstation [Online]. Available from: http://www.vmware.com/products/workstation/overview.html [2011, August 31].

[22] Free Software Foundation. GNU Parted [Online]. Available from: http://www.gnu.org/s/parted/ [2011, August 31].

ประวัติผู้เขียนวิทยานิพนธ์

นายวีรภัทร พรหมชนะ เกิดเมื่อวันที่ 14 ธันวาคม พ.ศ. 2528 ที่จังหวัดกรุงเทพหานคร สำเร็จการศึกษาระดับปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมสิ่งแวดล้อม จากคณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ ในปีการศึกษา 2549 และ เข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ ณ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2550 งานวิจัยที่สนใจ ได้แก่ ระบบเครือข่ายคอมพิวตอร์ และเทคโนโลยีเกี่ยวกับการบันทึกข้อมูล