

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

ระบบประเมินประสิทธิภาพระบบไฟล์แบบขนานพีวีเอฟเอส

บนระบบพีซีคลัสเตอร์

Performance Evaluation Tools for PVFS Parallel File System

on Commodities PC Cluster



611709076
112847118



รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 2 ปีการศึกษา 2547

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบรับรองโครงการพัฒนาระบบงาน (System Development Project)

เรื่อง

ระบบประเมินประสิทธิภาพระบบไฟล์แบบขนานพีวีเอฟเอส บนระบบพีซีคลัสเตอร์

Performance Evaluation Tools for PVFS Parallel File System

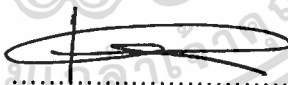
on Commodities PC Cluster

ว่าที่ร้อยตรีกิตติกริช ม่วงมิ่งสุข

รหัส 44067487

รายงานฉบับนี้ได้รับการตรวจสอบและอนุมัติให้เป็นส่วนหนึ่งของการศึกษาวิชาโครงการพัฒนาระบบงาน หลักสูตรวิทยาศาสตร์มหาบัณฑิต (เทคโนโลยีสารสนเทศ) ภาควิชาเทคโนโลยีพระจอมเกล้าธนบุรี

ภาคเรียนที่ 2 ปีการศึกษา 2547



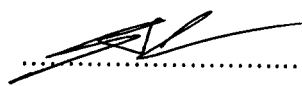
.....อาจารย์ที่ปรึกษา

(ผศ.ดร. วรพจน์ กวีสุระเดช)



.....กรรมการสอบ

(รศ.ดร. อาริต ธรรมโน)



.....กรรมการสอบ

(ผศ.ดร. โชติพัทธ์ ภรณวลัย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	ระบบประเมินประสิทธิภาพระบบไฟล์แบบขนานพีวีเอฟเอส บนระบบพีซีคลัสเตอร์
นักศึกษา	ว่าที่ร้อยตรีกิตติกร ม่วงมิ่งสุข
อาจารย์ที่ปรึกษา	ผศ. ดร. วรพจน์ กรีสระเดช
ระดับการศึกษา	วิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2547

บทคัดย่อ

ระบบพีซีคลัสเตอร์เป็นระบบขนาดใหญ่ ประกอบด้วยเครื่องคอมพิวเตอร์จำนวนมากช่วยกันทำงานทำให้มีขีดความสามารถในการคำนวณที่สูง แต่ยังมีส่วนที่จำกัดขีดความสามารถของระบบและถูกกล่าวว่าเป็นส่วนที่ช้าที่สุดของระบบนั่นก็คือส่วนของระบบอินพุทเอาต์พุท โดยเฉพาะดิสก์ ระบบไฟล์แบบขนานพีวีเอฟเอสเป็นระบบที่จะมาเพิ่มประสิทธิภาพทางด้านอินพุทเอาต์พุทให้กับระบบคลัสเตอร์โดยรวม ด้วยวิธีการกระจายภาระงานการให้บริการดิสก์ไปยังโหนดต่างๆ ในคลัสเตอร์ การที่จะออกแบบระบบคลัสเตอร์และกำหนดจำนวนเครื่องที่จะทำหน้าที่เป็นโหนดที่ให้บริการทางด้านอินพุทเอาต์พุทให้เหมาะสมนั้น จำเป็นต้องมีการประเมินประสิทธิภาพของระบบซึ่งระบบคลัสเตอร์เป็นระบบขนาดใหญ่มีความซับซ้อน การจะปรับแต่งระบบเพื่อประเมินประสิทธิภาพในหลากหลายรูปแบบกระทำได้ยาก โครงการพัฒนาระบบงานนี้นำเสนอระบบประเมินประสิทธิภาพของระบบไฟล์แบบขนานพีวีเอฟเอสบนระบบพีซีคลัสเตอร์ โดยนำหลักการของแถวคอยเครือข่ายมาประยุกต์ใช้ในการจำลองเหตุการณ์ เพื่อประเมินประสิทธิภาพของระบบซึ่งช่วยให้เกิดความสะดวกในการประเมิน ใช้เวลาสั้นลง และมีต้นทุนในการดำเนินงานที่ต่ำกว่า

Thesis Title	Performance Evaluation Tools for PVFS Parallel File System on Commodities PC Cluster
Student	Acting Sub, LT. Kittirak Moungmingsuk
Advisor	Asst. Prof. Dr. Worapoj Kreesuradej
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2004

ABSTRACT

Linux clusters of commodity computer systems and interconnects have become the fastest growing choice for building cost-effective high-performance parallel computing systems. The Parallel Virtual File System (PVFS) could potentially fulfill the requirements of large I/O-intensive parallel applications. It provides a high-performance parallel file system by striping file data across multiple cluster nodes, called I/O nodes. Thus, the performance of parallel I/O primitives is critical for overall cluster performance. The performance evaluation tools for PVFS parallel file system on commodities PC cluster has been developed. Our goal is to find the optimized PVFS performance. This tools can help PVFS users to obtain better performance from the system.

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานเรื่อง ระบบประเมินประสิทธิภาพระบบไฟล์แบบขนานพีวี
เอฟเอส นี้สำเร็จลุล่วงลงได้ด้วยคำแนะนำจาก ผศ.ดร.วรพจน์ กรีสระเดช อาจารย์ที่ปรึกษาประจำ
โครงการ ผู้เขียนต้องขอขอบพระคุณ

รศ. ดร. วิเชียร เปรมชัยสวัสดิ์ อาจารย์ที่ปรึกษาท่านแรกสำหรับงานด้านการประมวลผล
แบบขนาน

ผศ.ดร. ภูซงค์ อุทโยภาส หัวหน้าศูนย์วิจัยคอมพิวเตอร์สมรรถนะสูงและเครือข่าย
คอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ ผู้ชี้แนะแนวทางในการทำวิจัย และ
ให้ความกรุณาตลอดมา

ขอขอบคุณพี่ๆ เพื่อนๆ ในศูนย์วิจัยฯ ทุกท่าน ที่ให้คำแนะนำและเป็นกำลังใจให้
ขอขอบคุณมหาวิทยาลัยเกษตรศาสตร์ ที่เอื้อเพื่อให้ใช้ระบบคลัสเตอร์ในการทดลอง
ขอขอบคุณ คุณพ่อ คุณแม่ ที่คอยให้กำลังใจและสนับสนุนการเรียนเป็นอย่างดี
ขอขอบคุณทุกๆ ท่านที่ยัง ไม่ได้เอียนนาม ที่มีส่วนร่วมให้งานชิ้นนี้สำเร็จลุล่วงออกมา

กิตติรักษ์ ม่วงมิ่งสุข

มีนาคม 2548

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญรูป.....	VI
คำนิยามศัพท์.....	VII
บทที่	
1. บทนำ.....	1
1.1 ความเป็นมาของโครงการ.....	1
1.2 วัตถุประสงค์.....	2
1.3 ขอบเขตของการศึกษา.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
2. หลักการและทฤษฎีที่เกี่ยวข้อง.....	3
2.1 ระบบพีซีคลัสเตอร์.....	3
2.2 ระบบไฟล์แบบขนานพีวีเอฟเอส.....	7
2.3 การประเมินประสิทธิภาพทางคอมพิวเตอร์.....	10
2.4 ไลบารีจำลองเหตุการณ์แบบไม่ต่อเนื่องไฮเปอร์ซิม (HyperSim).....	15
3. วิธีดำเนินการศึกษา.....	17
3.1 สร้างแบบจำลอง.....	17
3.2 แบบจำลองของพีวีเอฟเอส.....	20
3.3 ตัวแบบพีวีเอฟเอส.....	21
3.4 ขั้นตอนการดำเนินงาน.....	22
3.5 การออกแบบโปรแกรมจำลอง.....	23
4. ผลการทดลอง.....	24
4.1 ระบบที่พัฒนา.....	24

สารบัญ (ต่อ)

	หน้า
4.2 ส่วนการทำงานของโปรแกรมประยุกต์.....	26
4.3 ผลการทดลอง.....	27
5. บทสรุปผลและข้อเสนอแนะ.....	29
5.1 สรุปผลโครงการ.....	29
5.2 ข้อเสนอแนะ.....	29
บรรณานุกรม.....	29
ประวัติผู้เขียน.....	31



สารบัญรูป

หน้า

รูปที่

2.1	สถาปัตยกรรมการเข้าถึงหน่วยความจำแบบใช้หน่วยความจำร่วมกัน	5
2.2	สถาปัตยกรรมการเข้าถึงหน่วยความจำแบบใช้หน่วยความจำร่วมกัน	5
2.3	ลักษณะการกระจายไฟล์ในระบบ	7
2.4	ลำดับชั้นการเชื่อมต่อระบบไฟล์แบบเสมือน	8
2.5	ภาพรวมโครงสร้างพีวีเอฟเอส.....	8
2.6	การติดต่อ MGR เพื่อขอ Metadata.....	9
2.7	การติดต่อโดยตรงระหว่าง APIs กับ I/O node	9
2.8	กระบวนการทำงานของไฮเปอร์ซีม.....	16
3.1	ตัวแบบเหตุการณ์ไม่ต่อเนื่องของพีวีเอฟเอส	17
3.2	ตัวแบบแถวคอยของพีวีเอฟเอส	18
4.3	กราฟแสดงประสิทธิภาพของระบบพีวีเอฟเอส บนเครือข่ายแบบฟาสต์อีเทอร์เน็ต	27
4.4	กราฟแสดงประสิทธิภาพของระบบพีวีเอฟเอส บนเครือข่ายแบบกิกาบิตอีเทอร์เน็ต.....	27

คำนิยามศัพท์

$t_{service}$	เวลาสำหรับการใช้บริการระบบพีวีเอฟเอส
$t_{getmeta}$	เวลาที่ใช้ในการร้องขอเมทาดต้า
t_{spread}	เวลาที่ใช้ในการกระจายหรือรวมไฟล์
$t_{transfer}$	เวลาที่ใช้ในการถ่ายโอนไฟล์ระหว่างลูกข่ายกับไอโอโนด
t_{disk}	เวลาที่ใช้ในการอ่านหรือเขียนไฟล์ของไอโอโนด
$t_{request}$	เวลาที่ใช้ในการร้องขอเมทาดต้า (Metadata)
$t_{metaservice}$	เวลาที่ใช้ในการจัดการเมทาดต้า
t_{reply}	เวลาที่ใช้ในการตอบเมทาดต้ากลับยังลูกข่าย
$t_{totaldelay}$	เวลาที่สูญเสียไประหว่างการถ่ายโอนข้อมูล
$t_{overhead}$	เวลาที่ใช้ในการเตรียมการก่อนการอ่านเขียนข้อมูลกับไฟล์
B_{agg}	ผลรวมแบนด์วิธ (Aggregate Bandwidth)
cp_n	จำนวนเครื่องลูกข่าย
ds	ขนาดของข้อมูล
B	แบนด์วิธ
S	อัตราการให้บริการของดิสก์
คลัสเตอร์ (Cluster)	ระบบคอมพิวเตอร์ที่สร้างจากการนำเอาเครื่องคอมพิวเตอร์หลายเครื่องเข้ามาเชื่อมต่อกันผ่านระบบเครือข่าย
โนด (Node)	คำเรียกเครื่องคอมพิวเตอร์ในคลัสเตอร์

บทที่ 1

บทนำ

1.1 ความเป็นมาของโครงการ

ในระบบคอมพิวเตอร์สมรรถนะสูงประเภทที่ใช้เครื่องคอมพิวเตอร์หลายๆ เครื่อง มาเชื่อมต่อกันเพื่อช่วยกันทำงานหรือที่เราเรียกว่า “คลัสเตอร์” นั้น การที่เราจะออกแบบระบบคลัสเตอร์ให้เหมาะสมกับงาน หรือให้ระบบมีขีดความสามารถตรงกับความต้องการสูงสุดนั้น จำเป็นต้องมีการทดสอบและวัดประสิทธิภาพของระบบก่อน แต่ก็จะต้องพิจารณาในด้านของงบประมาณ เนื่องจากระบบคลัสเตอร์อาจประกอบด้วยเครื่องคอมพิวเตอร์จำนวนมาก ซึ่งต้องใช้งบประมาณที่สูงในการจัดซื้อ ฉะนั้นการจัดซื้อมาเพื่อทดสอบคงเป็นไปได้ยาก ทางเลือกอีกทางที่จะช่วยให้เราสามารถประเมินประสิทธิภาพของระบบคลัสเตอร์ได้ก็คือการจำลอง

ในปัจจุบันมีการประยุกต์ใช้งานคลัสเตอร์กับระบบงานด้านต่างๆ จำนวนมาก ทั้งด้านที่ใช้ประโยชน์จากความสามารถในการประมวลผล การทนต่อภาวะการผิดพลาด หรือแม้แต่ความสามารถในการรองรับภาระงานจำนวนมาก ในโครงการนี้ มุ่งเน้นในด้านของการใช้ประสิทธิภาพจากระบบอินพุทเอาต์พุทของระบบการจัดเก็บข้อมูลบนดิสก์ การเชื่อมต่อและ

ระบบไฟล์แบบขนานพีวีเอฟเอสนี้เป็นระบบที่ทำงานอยู่บนระบบคลัสเตอร์ที่ประกอบด้วยเครื่องคอมพิวเตอร์หลายเครื่องเชื่อมต่อกัน ซึ่งอาจเชื่อมต่อกันได้หลายรูปแบบ แต่ละรูปแบบก็ให้ประสิทธิภาพที่แตกต่างกัน การที่จะทราบได้ว่าการปรับแต่งคลัสเตอร์แบบใดระบบถึงจะให้ประสิทธิภาพสูงสุดหรือตรงกับความต้องการที่สุดนั้น เราจำเป็นต้องทำการประเมินหรือวิเคราะห์ประสิทธิภาพของระบบ เพื่อให้ทราบถึงประสิทธิภาพของแต่ละสถาปัตยกรรมว่าระบบคลัสเตอร์ในแต่ละรูปแบบ จะให้ประสิทธิภาพมากน้อยต่างกันอย่างไร เพราะในระบบไฟล์แบบขนานนี้เป็นการทำงานของหลายส่วนประกอบกัน ส่วนประกอบหลักมีอยู่ 2 ส่วนคือ ส่วนของระบบเครือข่าย และส่วนของระบบดิสก์ ทั้งคู่เป็นปัจจัยหลักที่ส่งผลต่อประสิทธิภาพโดยรวมของระบบ ในระบบขนาดใหญ่ที่ประกอบด้วยปัจจัยจำนวนมาก คงจะเป็นการยากที่สามารถวิเคราะห์หาจุดที่ทำให้เกิดปัญหาคอขวด ซึ่งเป็นจุดที่ทำให้ไม่สามารถใช้งานระบบได้เต็มประสิทธิภาพ

1.2 วัตถุประสงค์

การศึกษาโครงการนี้มีจุดประสงค์เพื่อ

- ศึกษาพฤติกรรมการทำงานของระบบพีวีเอฟเอส ประสิทธิภาพและขีดความสามารถของระบบพีวีเอฟเอสบนระบบคลัสเตอร์
- พัฒนาโปรแกรมสำหรับประเมินประสิทธิภาพ (Performance Evaluation) ของระบบไฟล์แบบขนานพีวีเอฟเอส
- เพื่อศึกษาการทำงานและวิธีการใช้งานของระบบไฟล์แบบขนานพีวีเอฟเอส
- เพื่อศึกษาวิธีการประเมินสมรรถนะทางคอมพิวเตอร์ด้วยวิธีการจำลอง

1.3 ขอบเขตของการศึกษา

การศึกษาในโครงการนี้กำหนดขอบเขตในการศึกษาเป็นการพัฒนาระบบประเมินประสิทธิภาพระบบไฟล์แบบขนานพีวีเอฟเอส ด้วยวิธีการจำลอง เพื่อเปรียบเทียบประสิทธิภาพการทำงานของเครื่องปรับแต่งคลัสเตอร์ในแต่ละรูปแบบว่าแบบไหนจะให้ประสิทธิภาพการทำงานสูงสุด โดยมุ่งเน้นที่สองตัวแปรหลักคือ ความเร็วในการส่งผ่านข้อมูลของดิสก์และของเครือข่าย กับจำนวนเครื่องที่ให้บริการและเครื่องที่เข้าใช้บริการ โดยข้อมูลเบื้องต้นที่ใช้ในการทดสอบเป็นค่าที่ได้จากวัดจริงจากระบบคลัสเตอร์ที่มหาวิทยาลัยเกษตรศาสตร์ โดยมีขอบเขตหลักๆ ที่จะศึกษาคงต่อไปนี้

- ระบบไฟล์แบบขนานพีวีเอฟเอส
- ระบบคลัสเตอร์
- การจำลองสถานการณ์แบบไม่ต่อเนื่อง
- การประเมินประสิทธิภาพทางคอมพิวเตอร์

1.4 ประโยชน์ที่คาดว่าจะได้รับ

ผลจากการศึกษา จะทำให้เกิดความรู้ความเข้าใจหลักการทำงานของระบบไฟล์แบบขนานบนระบบพีซีคลัสเตอร์ พฤติกรรมของระบบไฟล์แบบขนานในสภาพแวดล้อมต่างๆ กัน เทคนิคการจำลองสถานการณ์ด้วยเทคนิคคิวแบบแถวคอย (Queuing Network Model) ทั้งยังเป็นการฝึกวิธีการค้นหาหาข้อมูล การวิเคราะห์ข้อมูล การแก้ปัญหาและแนวทางในการประยุกต์เทคนิคในการจำลองสถานการณ์กับสถานการณ์ต่างๆ

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 ระบบพีซีคลัสเตอร์

ระบบพีซีคลัสเตอร์ หรือเรียกสั้นๆ ว่าระบบคลัสเตอร์ ในที่นี้หมายถึงระบบประมวลผลแบบกลุ่มที่ประกอบด้วยเครื่องคอมพิวเตอร์หลายๆ เครื่องเชื่อมต่อเข้าด้วยกันผ่านเครือข่ายความเร็วสูง เช่น ฟาสต์อีเทอร์เน็ต (Fast Ethernet) กิกะบิตอีเทอร์เน็ต (Gigabit Ethernet) หรือ มายริเน็ต (Myrinet)¹ เป็นต้น โดยมีการเชื่อมต่อในรูปแบบเดียวกับระบบเครือข่ายแบบแลน (LAN) แต่จะแตกต่างกันตรงที่ระบบคลัสเตอร์จะมีซอฟต์แวร์ที่ช่วยจัดการระบบ เช่น ซอฟต์แวร์ในการกระจายภาระงาน ช่วยกระจายภาระงานไปทำยังเครื่องต่างๆ พร้อมๆ กัน ทำให้ระบบสามารถรองรับงานขนาดใหญ่ได้ เป็นต้น

เทคโนโลยีคอมพิวเตอร์สมรรถนะสูงถูกพัฒนาขึ้น เพื่อตอบสนองความต้องการของระบบงานที่ต้องการใช้ขีดความสามารถในการประมวลผลที่สูง เพื่อให้ได้ผลลัพธ์ทันต่อการใช้งาน โดยมากจะเป็นงานทางด้านวิทยาศาสตร์ เช่น ระบบการพยากรณ์อากาศ ระบบการจำลอง (Simulation) ในด้านต่างๆ เป็นต้น งานเหล่านี้ต้องการเครื่องคอมพิวเตอร์ที่มีประสิทธิภาพที่สูงกว่าเครื่องคอมพิวเตอร์ส่วนบุคคลที่ใช้งานกันทั่วไป เช่น เครื่องคอมพิวเตอร์เมนเฟรม เป็นต้น แต่ด้วยปัจจัยด้านราคา ทำให้เกิดมีกลุ่มวิจัยและพัฒนาระบบคอมพิวเตอร์สมรรถนะสูง ในรูปแบบพีซีคลัสเตอร์ (PC Cluster) ขึ้น เพื่อให้สามารถได้พลังในการคำนวณที่สูง ในราคาที่ต่ำกว่าเครื่องขนาดใหญ่ ทั้งยังสามารถประยุกต์ใช้งานเองได้โดยง่าย นับเป็นเทคโนโลยีทางเลือกทดแทนที่มีบทบาทกับงานวิจัยทางวิทยาศาสตร์และงานที่ต้องการความสามารถในการคำนวณขั้นสูงเป็นอย่างมาก

ในการประยุกต์ใช้งานระบบพีซีคลัสเตอร์ อาจแบ่งได้เป็นหลายรูปแบบตามลักษณะของการใช้ ซึ่งจะโครงสร้างและสถาปัตยกรรมของระบบคลัสเตอร์ในแบบต่างๆ ซึ่งจะได้อธิบายถึงในหัวข้อถัดไป

¹ มายริเน็ต (Myrinet) เป็นสวิตชิง แบบหนึ่งที่มีความสามารถในการส่งผ่านข้อมูลในอัตราที่สูงกว่า กิกะบิตอีเทอร์เน็ต สามารถดูข้อมูลเพิ่มเติมได้ที่ <http://www.myri.com/myrinet/performance/>

2.1.1 สถาปัตยกรรมทางฮาร์ดแวร์ของการประมวลผลแบบขนาน

พื้นฐานของแบ่งประเภทสถาปัตยกรรมทางฮาร์ดแวร์ของการประมวลผลแบบขนาน ตามทฤษฎีของ Flynn จัดแบ่งออกเป็น 4 ประเภท ดังต่อไปนี้

1. Single Instruction, Single Data (SISD)

เป็นสถาปัตยกรรมการประมวลผลแบบพื้นฐานที่สุด ซึ่งในหนึ่งหน่วยเวลาประมวลผลของหน่วยประมวลผลกลาง จะมีเพียงหนึ่งชุดคำสั่งและหนึ่งชุดข้อมูลเท่านั้น ที่ทำในช่วงเวลาดังกล่าว การทำงานเป็นไปในลักษณะเรียงลำดับ ซึ่งคอมพิวเตอร์ส่วนบุคคลก็จัดอยู่ในชุดนี้

2. Single Instruction, Multiple Data (SIMD)

เป็นการประมวลผลแบบที่สามารถแบ่งข้อมูลออกไปประมวลผล โดยสามารถทำการประมวลผลชุดคำสั่งเดียวกันในหลายๆ หน่วยประมวลผลได้พร้อมกันในเวลาเดียวกัน ซึ่งจำเป็นต้องมีการซิงโครไนส์ เพื่อให้มั่นใจว่าในช่วงเวลาเดียวกันได้ใช้ชุดคำสั่งเดียวกันในการประมวลผล และต้องมีการตรวจสอบว่าในแต่ละเวลาที่รัน โปรแกรมเดียวกัน ต้องรันบนข้อมูลชุดเดียวกัน

3. Multiple Instruction, Single Data (MISD)

วิธีนี้ก็จะเป็นการประมวลผลของชุดข้อมูลเพียงชุดเดียว โดยแบ่งกันออกไปประมวลผล โดยทำในหลายๆ ชุดคำสั่ง ซึ่งไม่ค่อยพบว่ามีการใช้งานทำให้สถาปัตยกรรมอันนี้ ไม่ค่อยมีการพัฒนา

4. Multiple Instruction, Multiple Data (MIMD)

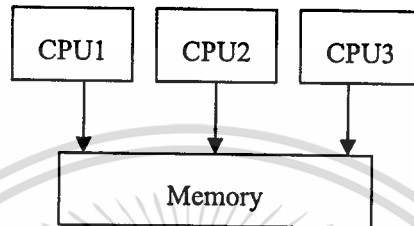
สำหรับวิธีนี้จัดว่าเป็นวิธีที่มีประสิทธิภาพสูงสุดตามแนวทางของการประมวลผลแบบขนาน ด้วยความสามารถในการแยกกันทำงานโดยรับข้อมูลมาหลายๆ ชุดข้อมูลและทำงานหลายๆ คำสั่ง ไปพร้อมๆ กัน ซึ่งมีทั้งแบบซิงโครไนส์และอะซิงโครไนส์ ซึ่งเราอาจแบ่งสถาปัตยกรรมแบบ MIMD นี้ ออกได้เป็น 2 ประเภทคือ

- ระบบหน่วยประมวลผลกลางหลายตัว คือเครื่องคอมพิวเตอร์ที่ถูกออกแบบให้มีหน่วยประมวลผลกลางได้มากกว่าหนึ่งตัว ซึ่งจะทำให้สามารถทำการประมวลได้เร็วขึ้นในรูปแบบนี้หน่วยประมวลผลจะใช้หน่วยความจำร่วมกัน
- ระบบคอมพิวเตอร์หลายเครื่อง คือการนำเครื่องคอมพิวเตอร์หลายๆ เครื่องมาเชื่อมต่อกันผ่านระบบเครือข่ายเพื่อให้ช่วยการประมวลผล ดังที่จะกล่าวต่อไปในหัวข้อ กลุ่มของเครื่องคอมพิวเตอร์ส่วนบุคคลสมรรถนะสูง

2.1.2 รูปแบบของการเข้าถึงหน่วยความจำ

1. ใช้หน่วยความจำร่วมกัน (Shared Memory)

รูปแบบของการเข้าถึงหน่วยความจำแบบนี้ จะพบได้ในเครื่องคอมพิวเตอร์ประเภทที่มีหน่วยประมวลผลหลายตัวในเครื่องเดียวกัน

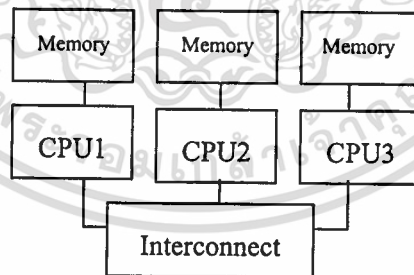


รูปที่ 2.1 สถาปัตยกรรมการเข้าถึงหน่วยความจำแบบใช้หน่วยความจำร่วมกัน

หลายๆ หน่วยประมวลผลใช้หน่วยความจำร่วมกันมีการชิงโครนัสการอ่านจากหน่วยความจำและเขียนลงหน่วยความจำรูปแบบนี้ใช้ในระบบหลายหน่วยประมวลผลแบบสมมาตร

2. กระจายหน่วยความจำ (Distributed Memory)

รูปแบบนี้จะพบในสถาปัตยกรรมที่ใช้เครื่องคอมพิวเตอร์เชื่อมต่อกันผ่านระบบเครือข่าย ซึ่งจะทำให้แต่ละหน่วยประมวลผลใช้หน่วยความจำแยกจากกันอย่างอิสระ



รูปที่ 2.2 สถาปัตยกรรมการเข้าถึงหน่วยความจำแบบใช้หน่วยความจำร่วมกัน

ข้อดีของรูปแบบนี้คือหน่วยประมวลผลสามารถเข้าถึงหน่วยความจำที่ตนเชื่อมต่ออยู่ได้โดยตรงสามารถควบคุมการเคลื่อนย้ายข้อมูลระหว่างหน่วยประมวลผลด้วยการชิงโครนัส

2.1.3 ประเภทของระบบคลัสเตอร์แบ่งตามรูปแบบการเชื่อมต่อ

อาจแบ่งประเภทของระบบคลัสเตอร์ตามรูปแบบการเชื่อมต่อได้เป็น 2 ประเภท คือ

1. การเชื่อมต่อแบบเปิด (Open System)

ในรูปแบบนี้จะเข้าไปในลักษณะของการประมวลผลแบบกระจาย โดยมากจะพบในลักษณะระบบแบบหลายผู้ใช้หรือระบบที่ต้องการให้บริการแบบหลายๆ งานพร้อมกัน อาทิเช่น ระบบเว็บที่มีผู้เข้าเยี่ยมชมเป็นจำนวนมาก ซึ่งในการเชื่อมต่อแบบนี้ผู้ใช้จะสามารถติดต่อกับแต่ละเครื่องในระบบได้โดยตรง ซึ่งอาจจะมีปัญหาหรือข้อพิจารณาอื่นๆ ตามมา อาทิเช่น การจัดสรรหมายเลขที่อยู่อินเทอร์เน็ต ให้กับแต่ละเครื่องซึ่งจะทำให้ต้องใช้จำนวนหมายเลขไอพีแอดเดรสตามจำนวนเครื่องที่ให้บริการ เป็นต้น

2. การเชื่อมต่อแบบปิด (Close System)

ในรูปแบบนี้กลุ่มของเครื่องคอมพิวเตอร์ที่เชื่อมต่อกันจะถูกเชื่อมต่อกับเครือข่ายภายนอก ณ จุดๆ เดียวในลักษณะของเกตเวย์ โดยมากจะมีโหนดที่จะทำหน้าที่ในการรับและแบ่งงานให้กับโหนดต่างๆ ภายในกลุ่ม ซึ่งในรูปแบบนี้จะเหมาะกับการนำไปประยุกต์ใช้งานในลักษณะของการวิจัยหรือพัฒนาโปรแกรมประยุกต์ กล่าวคือเป็นการใช้ความสามารถของเครื่องให้ช่วยกันทำงานงานหนึ่งให้เสร็จ

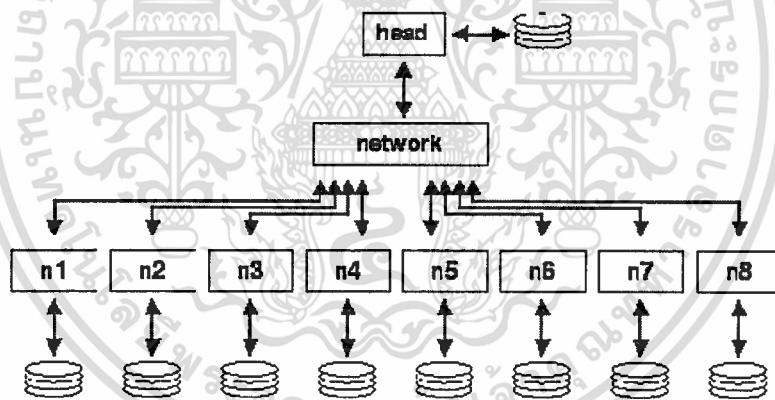
2.1.4 ระบบแนวฟูลคลัสเตอร์ (Beowulf Cluster)

เป็นระบบที่มีสถาปัตยกรรมเป็นแบบ MIMD ซึ่งสถาปัตยกรรมของคลัสเตอร์นี้ จะประกอบด้วยเครื่องพีซีคอมพิวเตอร์หลายๆ ตัวซึ่งเชื่อมต่อเข้าด้วยกันโดยใช้เครือข่ายความเร็วสูง เช่น อีเทอร์เน็ต ฟาสต์อีเธอร์เน็ต หรือ มายรีเน็ต เครื่องคอมพิวเตอร์แต่ละเครื่อง ที่นำมาต่อเชื่อมในระบบแนวฟูลคลัสเตอร์นี้จะเรียกว่าโหนดโดยจะมีฟรอนต์เอนด์โหนด เป็นโหนดหลักสำหรับทำการควบคุมระบบ และโหนดประมวลผลเป็นเครื่องลูกสำหรับทำการประมวลผลเป็นหลัก ในแต่ละโหนดจะใช้ระบบปฏิบัติการแบบ Full Multitask และจะสร้างภาพเสมือนว่าเป็นเครื่องเดียวกันหมดไม่ว่าผู้ใช้จะเข้าใช้งานเครื่องใดก็ตาม โดยใช้วิธีการต่างๆ เช่น ใช้ระบบไฟล์ NFS (Network File System) ในการสร้างระบบไฟล์ โดยมีเครื่องหนึ่งทำหน้าที่เป็นไฟล์เซอร์เวอร์สำหรับให้โหนดอื่นๆ นำเอาระบบไฟล์ไปใช้ หรือใช้ระบบ NIS (Network Information System) ในการกระจายข้อมูลเกี่ยวกับผู้ใช้ระบบไปยังทุกๆ เครื่อง เป็นต้น ภายในระบบจะมีซอฟต์แวร์สำหรับการประมวลผลแบบขนาน เช่น PVM หรือ MPI

2.2 ระบบไฟล์แบบขนานพีวีเอฟเอส

Carns et.al. (2000) ระบบไฟล์แบบขนานพีวีเอฟเอส (PVFS : Parallel Virtual File System) นี้ถูกคิดค้นและพัฒนาขึ้นโดยมหาวิทยาลัยเคลมสัน (Clemson University) ประเทศสหรัฐอเมริกา โดยวิวัฒนาการที่จะเพิ่มขีดความสามารถในการอ่านเขียนข้อมูลจากระบบดิสก์ด้วยความที่เป็นซอฟต์แวร์เสรี (Open source) ทำให้ระบบไฟล์แบบขนานพีวีเอฟเอสเป็นที่ได้รับความนิยมในการใช้งานบนระบบพีซีคลัสเตอร์

ระบบคลัสเตอร์ทำให้ขีดความสามารถในการประมวลผลเพิ่มสูงขึ้น ในขณะที่เดียวกันปริมาณการเรียกใช้งานข้อมูลก็ย่อมมีมากขึ้นไปตามกัน ทำให้ระบบคลัสเตอร์พบกับปัญหาการคับคั่งของข้อมูลเกิดเป็นคอขวดขึ้นในหลายที่ โดยเฉพาะการอ่านเขียนข้อมูลจากระบบไฟล์ เมื่อเครื่องคอมพิวเตอร์จำนวนหลายเครื่องติดต่อเพื่ออ่านเขียนข้อมูลจำนวนมากพร้อมๆ กัน ทำให้เกิดปัญหาความคับคั่งข้อมูลหรือคอขวดเกิดขึ้น แนวความคิดของระบบไฟล์แบบขนานจึงเกิดขึ้น เพื่อเข้ามาแก้ไขปัญหาดังกล่าวและเพื่อช่วยให้ประสิทธิภาพโดยรวมของระบบคลัสเตอร์สูงขึ้น

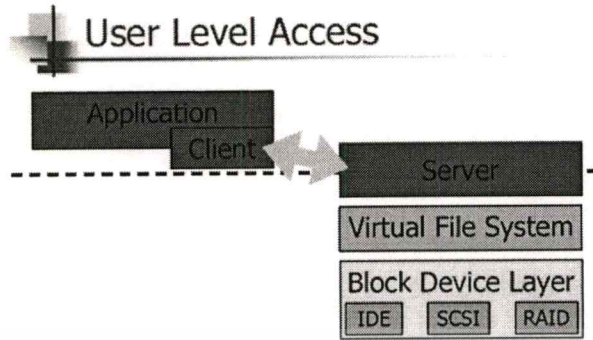


รูปที่ 2.3 ลักษณะการกระจายไฟล์ในระบบ

พีวีเอฟเอสใช้เทคนิคการตัดแบ่งไฟล์ออกเป็นบล็อกขนาดเล็กมีขนาดคงที่ โดยปกติใช้ขนาด 64 กิโลไบต์ แล้วกระจายกันออกไปจัดเก็บยังโหนดต่างๆ ในคลัสเตอร์ การกระจายไฟล์ไปจัดเก็บจะใช้วิธีการวนรอบ (Round Robin) (Ligon, 1996) แล้วกระจายกันนำไปจัดเก็บยังโหนดต่างๆ คลัสเตอร์ ดังรูปที่ 2.3 (Ross, 2002) แสดงให้เห็นถึงการกระจายไฟล์ไปจัดเก็บยังโหนดจำนวน 8 เครื่องด้วยกัน ในระบบคลัสเตอร์จะเรียกโหนดที่เก็บข้อมูลนี้ว่า ไอโอโหนด (I/O node)

ในมุมมองของผู้ใช้ภายนอกระบบไฟล์แบบขนานพีวีเอฟเอสจะสร้างภาพเสมือนให้กับระบบคลัสเตอร์ ทำให้ดูเหมือนเป็นที่เก็บข้อมูลขนาดใหญ่ ด้วยการสร้างดิสก์เสมือนโดยการรวมพื้นที่ของดิสก์บนเครื่องคอมพิวเตอร์หลายๆ เครื่อง ในคลัสเตอร์เข้ามาเสมือนเป็นชิ้นเดียวกัน

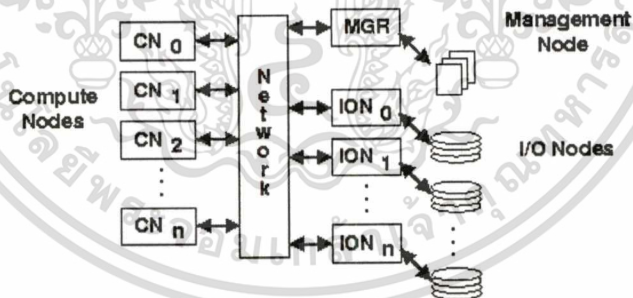
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 ลำดับชั้นการเชื่อมต่อระบบไฟล์แบบเสมือน

บนระบบไฟล์แบบขนานพวีเอฟเอส เราสามารถรันคำสั่งพื้นฐานของระบบยูนิกซ์และลินุกซ์ทั่วไปบนพวีเอฟเอสได้ กล่าวคือเราสามารถนำโปรแกรมที่พัฒนาภายใต้สภาพแวดล้อมของยูนิกซ์ มารันบนพวีเอฟเอส ได้โดยไม่ต้องคอมไพล์ใหม่ ด้วยเหตุที่พวีเอฟเอสทำงานอยู่เหนือระบบไฟล์ปกติของเครื่องแต่ละเครื่องในคลัสเตอร์ทำให้ระบบนี้ถูกเรียกว่าเป็นระบบไฟล์แบบเสมือน (Virtual File System) ลักษณะการเชื่อมต่อแสดงได้ดังรูปที่ 2.4

2.2.1 โครงสร้างของระบบไฟล์แบบขนานพวีเอฟเอส



รูปที่ 2.5 ภาพรวมโครงสร้างพวีเอฟเอส

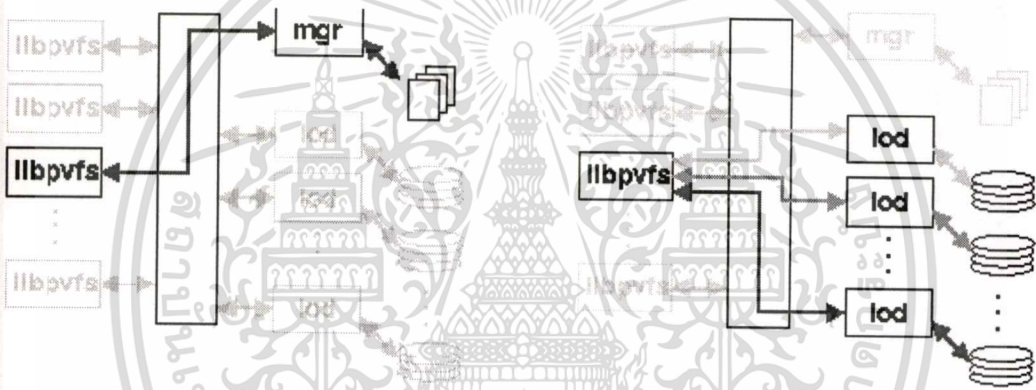
จากรูปที่ 2.5 (Ross, 2002) จะเห็นได้ว่าในระบบไฟล์แบบขนานพวีเอฟเอส ประกอบด้วยองค์ประกอบ 3 ส่วนหลัก คือ

1. โหนดบริหาร (Management Node) ทำหน้าที่จัดการ Metadata ซึ่งก็คือข้อมูลต่างๆ เกี่ยวกับไฟล์ที่จัดเก็บในระบบ เช่น ตำแหน่งของไฟล์ สิทธิของไฟล์ สถานะของการใช้ไฟล์ ประกอบด้วยการเปิดไฟล์ ปิดไฟล์ สร้างไฟล์ ลบไฟล์ แต่ไม่รวมถึงการอ่านและเขียนไฟล์ข้อมูล ด้วยลักษณะของงานที่ mgr รับผิดชอบนั้นเป็นสภาพบังคับให้ในระบบสามารถมี manager ได้เพียงหนึ่งเดียวเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2. ไอโอโหนด (I/O Nodes) เป็นส่วนที่จัดเก็บข้อมูลในระบบไฟล์แบบขนานพีวีเอฟเอส ไอโอโหนดจะรับไฟล์ที่ถูกตัดแบ่ง โดยนำมาจัดเก็บลงบนระบบไฟล์แบบปรกติของระบบ เช่น Ext2² เป็นต้น ทั้งยังทำหน้าที่ในการรับส่งไฟล์ข้อมูลให้กับเครื่องที่ร้องขอด้วย
- 3. โหนดประมวลผล (Compute node) เป็นผู้ใช้งานของระบบ ในการใช้งานระบบไฟล์แบบขนานพีวีเอฟเอสนี้จะมีขั้นตอนแตกต่างกับการใช้งานระบบไฟล์ทั่วไปโดยจะอธิบายในหัวข้อถัดไป

2.2.2 การทำงานของระบบไฟล์แบบขนานพีวีเอฟเอส



รูปที่ 2.6 การติดต่อ MGR เพื่อขอ Metadata

รูปที่ 2.7 การติดต่อโดยตรงระหว่าง APIs กับ I/O node

การทำงานจะเริ่มที่ฝั่งลูกข่ายหรือฝั่งโหนดประมวลผล ติดต่อไปยังโหนดบริหารเพื่อเพื่อร้องขอบริการไฟล์ ดังรูปที่ 2.6 Ross. (2002) โหนดบริหารจะให้ข้อมูลที่เกี่ยวข้องกับไฟล์ที่ต้องการ เช่น ตำแหน่งที่ไฟล์นั้นจัดอยู่ จะแจ้งให้ทราบถึงข้อมูลว่าถ้าต้องการไฟล์เหล่านี้จะต้องติดต่อกับไอโอโหนดโหนดใดบ้างที่ส่วนไหน เป็นต้น จากนั้นเป็นหน้าที่ของลูกข่ายเองที่จะต้องร้องขอและกระทำการอ่านเขียนข้อมูลโดยตรงกับไอโอโหนด ดังรูปที่ 2.7 Ross. (2002) กระบวนการติดต่อสื่อสารทั้งหมดจะใช้ TCP เป็นโปรโตคอลในการติดต่อ

เอกสาร² EXT2 ย่อมาจาก Second Extended File system เป็นระบบไฟล์บนลินุกซ์แบบหนึ่ง ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การประเมินประสิทธิภาพทางคอมพิวเตอร์

วรา วราวิทย์. (2545) กล่าวว่า การวิเคราะห์ประสิทธิภาพของระบบคอมพิวเตอร์สามารถแบ่งได้เป็น 3 เทคนิคหลักได้แก่ การวัดจากระบบจริง การจำลอง และการวิเคราะห์ทางคณิตศาสตร์ ซึ่งแต่ละเทคนิคก็มีข้อดีข้อเสียต่างกัน การวัดจากระบบจริง ดูแล้วน่าจะเป็นวิธีที่ดีที่สุด มีความน่าเชื่อถือสูง แต่ในบางกรณีเราไม่อาจสามารถวัดจากระบบจริงได้ด้วยเหตุผลบางประการ การจำลองจะเข้ามาทดแทนตรงจุดนี้ได้เป็นอย่างดี

การคำนวณหาประสิทธิภาพของระบบคลัสเตอร์ว่ามีประสิทธิภาพเท่าใดเหมาะสมกับงานตามชนิดประเภทที่ต้องการใช้งานหรือไม่ ในทางปฏิบัติถ้าจะหาคำตอบจากปัญหาดังกล่าว อาจต้องทดสอบในหลากหลายรูปแบบ จำเป็นต้องใช้ฮาร์ดแวร์จำนวนมากเพื่อสร้างเป็นตัวแบบและวัดประสิทธิภาพของระบบ ซึ่งจำเป็นต้องใช้งบประมาณที่สูง การจำลองจะช่วยให้สามารถทำได้โดยมีค่าใช้จ่ายที่ต่ำกว่า และในระบบการทำงานที่มีขั้นตอนซับซ้อนมากๆ การจำลองสถานการณ์จะช่วยให้เราเห็นภาพและเข้าใจถึงลำดับการเกิดขึ้นของเหตุการณ์ได้สะดวกขึ้น สำหรับระบบไฟล์แบบขนานพีวีเอฟเอสนี้ มีขั้นตอนในการทำงานที่ยุ่งยากซับซ้อน ดังนั้นในงานชิ้นนี้จึงเลือกใช้วิธีการจำลองเหตุการณ์ (Simulation Technique) ด้วยวิธีตัวแบบแถวคอย (Queuing Model) ในการประเมินประสิทธิภาพของระบบดังกล่าว

2.3.1 การจำลอง

ฤดี มาสุจันทร์ (2546) การจำลอง (Simulation) เป็นเทคนิคที่ใช้ศึกษาปัญหาที่มีความซับซ้อน โดยการเลียนแบบการทำงานจริงของกระบวนการในช่วงระยะเวลาหนึ่งๆ เพื่อศึกษาถึงคุณสมบัติต่างๆ ของกระบวนการ

การจำลองเหตุการณ์ เป็นการดำเนินการบนตัวแบบที่ถูกสร้างขึ้นเพื่อเป็นตัวแทนระบบจริงในช่วงหนึ่งๆ เพื่อศึกษาพฤติกรรมของระบบ และนำผลที่ได้จากตัวแบบไปใช้ในการพยากรณ์พฤติกรรมของระบบงานที่แท้จริง ในการจำลองเหตุการณ์จะใช้ข้อมูลเก่าหรือข้อมูลที่วางแผนไว้ก็ได้ โดยตัวแบบที่ใช้สามารถแสดงผลลัพธ์การคำนวณต่างๆ ได้

การจำลอง (Simulation) เป็นเทคนิคหนึ่งซึ่งช่วยให้เราสามารถศึกษาพฤติกรรมของระบบและสามารถประเมินประสิทธิภาพของระบบในสถานะต่างๆ ได้โดยง่าย ซึ่งการจำลองสถานการณ์ทางคอมพิวเตอร์ก็มีหลายเทคนิคที่นิยมใช้กัน ในงานนี้เราเลือกใช้เทคนิคที่เรียกว่าแถวคอยเครือข่าย (Queuing Network) มาประยุกต์ใช้ในการจำลองสถานการณ์ของระบบไฟล์แบบขนานพีวีเอฟเอส เพื่อจัดทำเป็นตัวแบบประสิทธิภาพ (Performance Model) ของระบบ ซึ่งจะช่วยให้ทราบถึงขีดความสามารถของระบบในสถานะต่างๆ ซึ่งจะช่วยให้สามารถออกแบบและพัฒนาเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบจริงให้มีประสิทธิภาพตรงกับความต้องการใช้งาน และใช้ความสามารถของระบบอย่างคุ้มค่า เต็มกำลัง การจำลองสามารถพยากรณ์ประสิทธิภาพของระบบในสถานะต่างๆ ตามเหตุการณ์ที่เกิดขึ้น การวัดจากระบบจริงไม่สามารถยืนยันได้ว่าในขณะนั้นระบบได้ให้ประสิทธิภาพสูงสุดตามที่ควรจะเป็น

การจำลองจะช่วยให้เราสามารถพยากรณ์ประสิทธิภาพและวิเคราะห์ปัญหาที่เกิดขึ้นกับระบบได้โดยง่ายเพราะเราสามารถปรับแต่งพารามิเตอร์ต่างๆ ได้ การใช้งานพารามิเตอร์ให้ใกล้เคียงกับความเป็นจริงและปรับแต่งพารามิเตอร์ให้สอดคล้องกับความเป็นจริง ย่อมส่งผลให้แบบจำลองนั้นมีความน่าเชื่อถือ และอาจแสดงถึงคุณลักษณะของระบบได้ใกล้เคียงกับระบบจริง

2.3.2 ตัวแบบแถวคอย (Queuing Model)

ตัวแบบแถวคอยเป็นเทคนิคการวิเคราะห์เชิงปริมาณแบบหนึ่งที่ใช้กันแพร่หลายมานาน ถูกนำมาใช้วิเคราะห์ระบบแถวคอยเพื่อการตัดสินใจ ในระบบแถวคอยมุ่งคำนวณหาจำนวนการรอคอยในระบบ เพื่อแก้ไขวิเคราะห์และแก้ปัญหาความสามารถในการให้บริการของหน่วยบริการต่อปริมาณหรืออัตราของผู้มาใช้บริการ เพื่อค้นหาคำตอบว่า ณ จุดใดถึงจะเหมาะสมหรือได้ผลตามวัตถุประสงค์ที่ตั้งไว้ ซึ่งวัตถุประสงค์ที่ตั้งไว้อาจต่างกัน เช่น ต้องการให้บริการโดยเร็วที่สุด ควรใช้เซิร์ฟเวอร์จำนวนเท่าใด หรือต้องการให้เกิดความเหมาะสมทั้งทางด้านค่าใช้จ่ายและความเร็วควรใช้เครื่องจำนวนเท่าใด เป็นต้น

ในชีวิตประจำวันของคนในปัจจุบันจะต้องเข้าไปมีส่วนร่วมในระบบแถวคอย หรือระบบคิว ในลักษณะใดลักษณะหนึ่ง เช่น รอขึ้นรถประจำทาง รอซื้ออาหาร เป็นต้น ดังนั้นจึงจำเป็นต้องมาทำความเข้าใจกับลักษณะ และการจัดระบบแถวคอยว่าจะเกิดขึ้นเมื่อมีผู้เข้ามาใช้บริการ ซึ่งเราเรียกว่า ผู้ใช้บริการ เข้ามาในระบบเพื่อรับบริการแต่ไม่ได้รับบริการในทันทีที่มาถึงเนื่องจากผู้ให้บริการ ซึ่งเรียกว่า หน่วยให้บริการกำลังให้บริการผู้ให้บริการรายอื่นอยู่จึงต้องรอ

ทฤษฎีแถวคอย ริเริ่มพัฒนาในปี พ.ศ. 2453 โดย เอ.เค.เออร์แลง วิศวกรชาวเดนมาร์ก เพื่อแก้ปัญหารอคอยของผู้ใช้โทรศัพท์ หลังจากนั้นก็มีผู้ทำการศึกษาและพัฒนาระบบแถวคอยอื่นๆ อีกมาก แต่อย่างไรก็ตามตัวแบบแถวคอยทุกตัวแบบมุ่งเน้นที่จะอธิบายหรือวิเคราะห์ระบบแถวคอยโดยแสดงข้อมูลตัวเลขต่างๆนอกเหนือจากการหาวิธีที่ดีที่สุดในการแก้ปัญหา

แถวคอยเกิดขึ้นเมื่อความต้องการรับบริการมีมากกว่าความสามารถในการให้บริการ ดังนั้นการจัดให้มีจำนวนหน่วยให้บริการที่เพียงพอจึงเป็นสิ่งสำคัญยิ่ง ถ้ามีจำนวนหน่วยให้บริการน้อยเกินไปจะเกิดแถวคอยซึ่งนับเป็นการสูญเสียค่าใช้จ่ายอย่างหนึ่ง และอาจทำให้เสียผู้ให้บริการด้วย แต่ถ้ากิจการจัดให้มีหน่วยให้บริการมากเกินไปก็จะทำให้เกิดค่าใช้จ่ายเกินจำเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนนี้จะอธิบายถึง โครงสร้างทั่วไปของระบบแถวคอย ลักษณะต่างๆของระบบแถวคอยที่จำเป็นต้องทราบ การวิเคราะห์ตัวแบบแถวคอย โดยจะกล่าวในหัวข้อถัดๆ ไป

2.3.3 โครงสร้างของระบบแถวคอย

ระบบแถวคอยต่างๆไป จะประกอบด้วยส่วนประกอบที่สำคัญ 2 ส่วน คือ

- 1) แถวคอย
- 2) หน่วยให้บริการ

2.3.4 ระบบแถวคอย

1. รูปแบบของระบบ มีอยู่ 4 แบบคือ

1.1) ระบบแถวคอยช่องทางเดียว-ขั้นตอนเดียว คือ ระบบแถวคอยที่มีหน่วยบริการหน่วยเดียวและมีขั้นตอนการบริการขั้นตอนเดียว

1.2) ระบบแถวคอยช่องทางเดียว-หลายขั้นตอน คือ ระบบแถวคอยที่มีขั้นตอนการบริการหลายขั้นตอน และแต่ละขั้นตอนมีหน่วยบริการหน่วยเดียว

1.3) ระบบแถวคอยหลายช่องทาง-ขั้นตอนเดียว คือ ระบบแถวคอยที่มีขั้นตอนการบริการขั้นตอนเดียวแต่มีหลายหน่วยให้บริการ

1.4) ระบบแถวคอยหลายช่องทางหลายขั้นตอน คือระบบแถวคอยที่มีขั้นตอนการบริการหลายขั้นตอนและแต่ละขั้นตอนมีหลายหน่วยบริการ

2. รูปแบบการให้บริการ (Service Discipline):

รูปแบบการให้บริการหมายถึงกฎเกณฑ์ที่ระบบนั้นใช้ในการกำหนดว่าจะให้บริการผู้ใช้บริการรายใดก่อน มีด้วยกันหลายรูปแบบ ที่นิยมใช้และพบบ่อย ได้แก่

- 2.1) ผู้ใช้บริการที่มาก่อนจะได้รับบริการก่อน (FCFS)
- 2.2) ผู้ใช้บริการที่มาทีหลังจะได้รับบริการก่อน (LCFS)

3. ความยาวของแถวคอย (Queue length)

- 3.1) ความยาวแบบจำกัด
- 3.2) ความยาวไม่จำกัด

4. ลักษณะของผู้ใช้บริการ

1. จำนวนประชากร (Population Size) คือผู้ที่มีโอกาสจะเข้ามาใช้บริการในระบบ ซึ่งในบางครั้งจะพบว่าผู้ที่สามารถจะเข้ามาในระบบได้นั้นมีมากมาย ซึ่งเรียกว่าประชากรไม่จำกัด แต่ระบบแถวคอยบางระบบจะมีผู้เข้ามาใช้บริการจำนวนน้อยมาก เรียกว่าจำนวนประชากรจำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ลักษณะการเข้ามาใช้บริการ (Arrival Process) ซึ่งลักษณะการเข้ามาใช้บริการของผู้ใช้บริการแบ่งเป็น 2 แบบคือ

2.1) การเข้ามาใช้บริการในอัตราที่คงที่ คือ . ผู้ใช้บริการเข้ามาในลักษณะสม่ำเสมอ

2.2) การเข้ามาใช้บริการแบบสุ่ม คือ ผู้ใช้บริการเข้ามาในลักษณะที่แน่นอน ไม่สม่ำเสมอ ไม่ทราบล่วงหน้า และการเข้ามาของผู้ใช้บริการแต่ละรายเป็นอิสระต่อกัน ซึ่งในบางเวลา มีผู้ให้บริการเข้ามากราย บางเวลาอาจมีน้องรายหรือไม่มีเลย ในการใช้เทคนิคเชิงปริมาณเพื่อวิเคราะห์ระบบแถวคอยนั้นๆจึงจำเป็นต้องใช้ค่าเฉลี่ยของการเข้ามาใช้บริการ โดยต้องศึกษาการแจกแจงความน่าจะเป็นของการเข้ามาใช้บริการด้วยว่าเป็นแบบใด เช่นแบบปกติ ,แบบปัวส์ซอง เป็นต้น

2.3.5 การเก็บข้อมูลการเข้ามาใช้บริการ

สามารถทำได้ 2 ลักษณะ

- เก็บข้อมูลในลักษณะอัตราการเข้ามาใช้บริการ คือผู้ให้บริการเข้ามาใช้บริการโดยเฉลี่ยกี่คนในหนึ่งหน่วยเวลา เช่นรถเข้ามาเติมน้ำมัน 10 คันต่อชั่วโมง

- เก็บข้อมูลในลักษณะเวลาระหว่างการเข้ามาใช้บริการ คือเวลาห่างโดยเฉลี่ยระหว่างผู้ให้บริการแต่ละคน

ระบบแถวคอยจะมีลักษณะการมารับบริการแบบสุ่ม โดยอัตราการเข้ามาใช้บริการมีการแจกแจงแบบปัวส์ซอง สมมติฐานแถวคอยคือ ผู้ใช้บริการจะรอจนกว่าจะได้รับบริการ

2.3.6 อัตราการให้บริการ (Service Time Distribution)

คือจำนวนผู้ให้บริการที่สามารถให้บริการได้ในหนึ่งหน่วยเวลา อัตราการเข้ามาใช้บริการแบ่งเป็น 2 ลักษณะ คือ

1. อัตราการให้บริการแบบคงที่ คือในการบริการผู้ให้บริการแต่ละรายใช้เวลาเท่าๆ กัน ดังนั้นทุกหน่วยเวลาก็จะให้บริการผู้ให้บริการได้จำนวนเท่ากันเสมอ

2. อัตราการให้บริการแบบสุ่ม คือการให้บริการผู้ให้บริการแต่ละรายให้บริการไม่เท่ากันมากบ้างน้อยบ้างตามความต้องการของผู้ใช้บริการ

ในการวิเคราะห์ระบบแถวคอยจึงใช้ค่าเฉลี่ยของการให้บริการ รวมถึงศึกษาถึงการแจกแจงของข้อมูลการให้บริการด้วย การเก็บข้อมูลในด้านการให้บริการมักอยู่ในรูปของเวลาที่ใช้ในการบริการ มากกว่าอัตราการให้บริการ คือจะบันทึกเวลาที่ใช้ในการให้บริการผู้ให้บริการแต่ละราย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วนำมาหาค่าเฉลี่ย ระบบแถวคอยส่วยใหญ่จะมีการให้บริการแบบสุ่ม และมีการแจกแจงแบบเอกโพเนนเชียล

2.3.7 สัญลักษณ์ที่ใช้ในตัวแบบแถวคอย

วรา วราวิทย์. (2545) กล่าวว่า ดี.จี.เคนคอลล นักคณิตศาสตร์ชาวอังกฤษ ได้ออกแบบการ แสดงลักษณะของตัวแบบแถวคอย เรียกว่า เคนคอลลโนเตชัน ดังต่อไปนี้

A / S / m / B / K / SD

โดยที่แต่ละตัวอักษรแสดงถึงพารามิเตอร์ต่างๆ ดังต่อไปนี้

- A = การแจกแจงของอัตราการมารับบริการ (Arrival Rate)
- S = การแจกแจงของอัตราการให้บริการ (Service Time Distributrion)
- m = จำนวนหน่วยผู้ให้บริการ (Number of Servers)
- B = ขนาดของระบบ (System Capacity)
- K = ขนาดประชากร (Population Size)
- SD = รูปแบบการบริการ (Service Discipline)

การแจกแจงของอัตราการมารับบริการ และการให้บริการใช้สัญลักษณ์ดังนี้

- M = แบบเอ็กโปเนนเชียล (Exponential)
- D = แบบคงที่ (Deterministic)
- G = แบบทั่วไป (General)

สังเกตว่าการกระจายแบบ Exponential แสดงโดยตัวอักษรย่อ M ที่ย่อมาจาก Memory less หรือไม่มีความจำ ถ้าระยะเวลาระหว่างการเข้าใช้ระบบมีการกระจายแบบ Exponential ด้วยค่าเฉลี่ย $1/\lambda$ ค่าคาดหวัง (ค่าเฉลี่ย) ของการเข้าใช้ระบบครั้งต่อไปจะเป็น $1/\lambda$ เสมอ ไม่ขึ้นอยู่กับเวลาการเข้าใช้ระบบในครั้งก่อนหน้า คุณสมบัติดังกล่าวเรียกว่า Memory less ข้อมูลการเข้าใช้ระบบในอดีตไม่สามารถช่วยในการคาดหวังการเข้าใช้ระบบในอนาคตได้

λ = อัตราการเข้ามารับบริการ (จำนวนผู้ใช้บริการโดยเฉลี่ยที่เข้ามารับบริการในหนึ่งหน่วยเวลา)

μ = อัตราการให้บริการ (จำนวนผู้ใช้บริการโดยเฉลี่ยที่หน่วยบริการให้บริการได้ในหนึ่งหน่วยเวลา)

2.3.8 กฎของลิตเติล (Little's Law)

จำนวนผู้ใช้บริการในระบบจะเท่ากับจำนวนผู้ใช้ระบบ โดยเฉลี่ยคูณด้วยจำนวนเวลา
รับบริการ

$$\text{Mean number tasks in system} = \text{arrival rate} \times \text{mean service time}$$

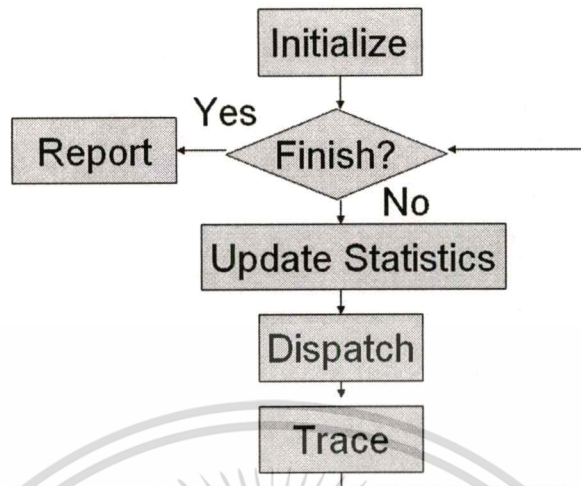
กฎดังกล่าวเป็นจริงก็ต่อเมื่อ จำนวนงานที่เข้าใช้ระบบเท่ากับจำนวนงานที่เสร็จการ
บริการจากระบบ ดังนั้นจะไม่มีงานที่ถูกสร้างขึ้นหรือเกิดขึ้นใหม่ในระบบ และงานที่สูญหายไป
จากระบบ

2.3.9 ตัวแบบเครือข่ายแถวคอย (Queuing Network Model)

ข้างต้นได้กล่าวถึงตัวแบบแถวคอย ซึ่งมีลักษณะที่มีหนึ่งแถวคอยและหนึ่งหน่วยบริการ
เท่านั้น ส่วนในระบบคอมพิวเตอร์จริงๆ จะมีการทำงานของหลายหน่วยบริการต่อเนื่องกัน
ประกอบด้วยหลายแถวคอย และหลายหน่วยให้บริการ มีการเข้าใช้บริการหนึ่งแล้วออกมาเข้า
แถวคอยเพื่อใช้บริการอื่นๆ ก่อนที่จะเสร็จสิ้นงาน เราจะเรียกตัวแบบลักษณะนี้ว่า เครือข่ายแถวคอย
จัดเป็นเทคนิคหนึ่งในการจำลอง ซึ่งเป็นใช้กับปัญหาที่มีความยุ่งยากซับซ้อน เช่น ประเภทที่มีหลาย
ตัวแปร เป็นต้น การจำลองด้วยระบบนี้จะเป็นการจำลองสภาพเหมือนกัน มีการกำหนดตัวแปร
ต่างๆ ให้มันทำงานไปตามวงจร และกำหนดเหตุการณ์ต่างๆ ใ้เข้าไป

2.4 ไอบารีจำลองเหตุการณ์แบบไม่ต่อเนื่องไฮเปอร์ซิม (HyperSim)

ไฮเปอร์ซิม (HyperSim) เป็นไลบรารีภาษาซีพลัสพลัส (C++) ที่พัฒนาร่วมกันระหว่าง
สถาบันเทคโนโลยีแห่งเอเชียและศูนย์วิจัยคอมพิวเตอร์สมรรถนะสูง มหาวิทยาลัยเกษตรศาสตร์
ภายในประกอบด้วยคลาสและฟังก์ชันที่สำคัญต่องานจำลองสถานการณ์ โดยเป็นการจำลองแบบ
เหตุการณ์ไม่ต่อเนื่อง (Discrete Event Simulation) Phatanapherom et.al (2003) ไฮเปอร์ซิม
ประกอบด้วยคิสตรีบิวชันมากมาย เช่น ปัวส์ซอง (Poisson) เป็นต้น ใช้หลักการของการเปลี่ยน
Stage ในการทำงานทำให้ช่วยลดความซับซ้อนของการพัฒนาโปรแกรมลง



รูปที่ 2.8 กระบวนการทำงานของไฮเปอร์ซิม

ไฮเปอร์ซิมมีกระบวนการในการทำงานดังรูปที่ 2.8 โดยจะเริ่มต้นที่ขั้น Initialize เป็นการสถานะของสถานการณ์ จากนั้นจะเป็นการตรวจสอบเงื่อนไขว่าการจำลองเหตุการณ์นี้สิ้นสุดหรือไม่ โดยโปรแกรมจะสิ้นสุดเมื่อไม่มีเหตุการณ์เหลืออยู่ในแถวคอย หรือเหตุการณ์เป็นไปตามเงื่อนไขที่ผู้ใช้กำหนด ถ้าการจำลองยังไม่สิ้นสุด โปรแกรมจะปรับค่าต่างๆ จัดเก็บไว้ในตัวแปร แล้วจึงเริ่มจัดตารางงานเข้าไปทำงานในสถานะต่างๆ ต่อไป ทำเช่นนี้เป็นวงรอบวนไปเรื่อยๆ จนกว่าจะจบ แล้วจึงนำค่าผลลัพธ์ได้ที่คำนวณเก็บไว้มาแสดง

ในส่วนของไลบรารีมีคลาสหลักในการทำงาน ดังแสดงได้ดังต่อไปนี้

```

class Simulation {
public:
    Simulation(Simulation &root);
    ~Simulation();
    virtual int init();
    virtual int dispatch(EventNode *node);
    virtual bool isFinish();
    virtual int updateStat();
    virtual void trace(EventNode *node);
};
  
```

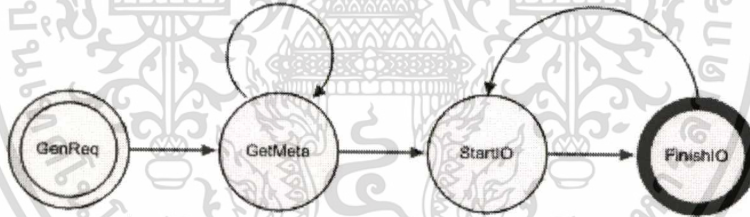
บทที่ 3

วิธีการดำเนินการศึกษา

ในการศึกษาโครงการนี้ เป็นการนำทฤษฎีตัวแบบแถวคอยมาพัฒนาเป็นโปรแกรมคอมพิวเตอร์ เพื่อประยุกต์ใช้ในการประเมินประสิทธิภาพของระบบไฟล์แบบขนานพีวีเอฟเอส โดยแบ่งการดำเนินการศึกษาออกเป็น 3 ส่วน ดังนี้

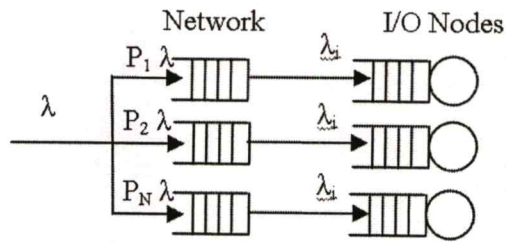
1. สร้างแบบจำลอง
2. ขั้นตอนการดำเนินงาน
3. การออกแบบโปรแกรมจำลอง

3.1 สร้างแบบจำลอง



รูปที่ 3.1 ตัวแบบเหตุการณ์ไม่ต่อเนื่องของพีวีเอฟเอส

จากรูปที่ 3.1 แสดงตัวแบบเหตุการณ์การทำงานของซอฟต์แวร์ประเมินประสิทธิภาพระบบไฟล์แบบขนานพีวีเอฟเอส โดยขั้นเริ่มต้นจะเริ่มเหตุการณ์ GenReq เป็นการจำลองสถานะการร้องขอเพื่ออ่านเขียนข้อมูลของผู้ใช้มายังระบบพีวีเอฟเอส เหตุการณ์ Getmeata แสดงถึงการที่โหนดประมวลผลร้องขอข้อมูลเมทาดัต้า จากโหนดบริหารซึ่งทำหน้าที่บริหารจัดการเมทาดัต้า เมื่อได้ข้อมูลตำแหน่งที่อยู่ของข้อมูลที่ต้องการแล้วจะเข้าสู่เหตุการณ์ StartIO แสดงถึงการเริ่มร้องขอไฟล์และรับส่งไฟล์จากไอโอโหนด ซึ่งจะทำเรื่อยๆ จนกว่าจะได้ข้อมูลครบตามที่ต้องการ ซึ่งเหตุการณ์ FisishIO จะเป็นส่วนที่ทำให้สัญญาณกระบวนการถ่ายโอนข้อมูลได้เสร็จสิ้นแล้ว



รูปที่ 3.2 ตัวแบบแถวคอยของพีวีเอฟเอส

แถวคอยของระบบเป็นประเด็นหลักที่เราต้องทำการวิเคราะห์ เนื่องจากระบบพีวีเอฟเอส ประกอบด้วยเครื่องคอมพิวเตอร์จำนวนมาก การออกแบบในส่วนของแถวคอยจึงมีบทบาทสำคัญต่อประสิทธิภาพของระบบโดยรวม โดยเราจะกำหนดให้มีแถวคอยอยู่ที่สองจุดหลักคือที่ระบบเครือข่าย และที่ระบบดิสก์ ดังรูปที่ 3.2 ปริมาณการเข้าแถวคอยจะถูกแบ่งโดยระบบไฟล์แบบขนานซึ่งจะใช้วิธีการตัดแบ่งเป็นบล็อกข้อมูลมีขนาดเท่าๆ กัน และกระจายด้วยวิธีแบบ Round Robin ไปยังทุกๆ แถวคอย ทำให้ปริมาณงานในแต่ละแถวคอยมีขนาดใกล้เคียงกันมาก ซึ่งการแบ่งแถวคอยออกเป็นตามรูปจะทำให้เห็นภาพการทำงานของระบบได้ชัดเจน

ในโครงการงานนี้ได้ประยุกต์ใช้ตัวแบบแถวคอยแบบ M/G/1 กับระบบดิสก์ และกระบวนการร้องขอของระบบอินพุทเอาต์พุทแทนด้วยค่าการกระจายพัลส์ของ ใช้ตาม Feng et.al. (2003) ด้วยอัตราปริมาณการเข้าใช้ระบบโดยเฉลี่ย λ ค่าการให้บริการของระบบเครือข่ายและระบบดิสก์ให้มีการกระจายแบบเอ็กซ์โปเนนเชียล ตัวแบบที่ใช้ในโครงการนี้เป็นลักษณะคิวแบบปิด มีลำดับการเข้าใช้งานเป็นแบบมาก่อนให้บริการก่อน (FCFS)

จากภาพที่ 3.1 แสดงถึงการทำงานของพีวีเอฟเอส ซึ่งประกอบด้วยการทำงานในขั้นตอนย่อยทั้งหมด ดังนี้

1. ชั้นร้องขอเมทาาดาต้า
2. ชั้นถ่ายโอนข้อมูล
3. ชั้นอ่านเขียนข้อมูล

โดยในแต่ละขั้นตอนมีรายละเอียดดังต่อไปนี้

3.1.1 ชั้นร้องขอเมทาาดาต้า

ก่อนที่เครื่องลูกข่ายจะทำการอ่านหรือเขียนข้อมูลจากระบบพีวีเอฟเอสได้นั้น เครื่องลูกข่ายจะต้องติดต่อกับพีวีเอฟเอสเมทาาดาต้าที่โหนดบริหารก่อน โดยติดต่อผ่านพีวีเอฟเอสไลบรารี เพื่อเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ร้องขอให้โหนดบริหารให้บริการเปิดปิดไฟล์ หรือสร้างและลบไฟล์ตามแต่ที่ความต้องการใช้บริการ หลังจากนั้นเครื่องเมทาดาต้าจะติดต่อไปยังไอโอโหนดทุกโหนด เพื่อแจ้งให้แต่ละไอโอโหนดเปิดรับการติดต่อหรือกระทำการตามคำร้องขอ จากนั้นโหนดบริหารจะส่งผลแจ้งถึงความพร้อมในการให้บริการให้กับเครื่องลูกข่ายที่ร้องขอ ซึ่งจะใช้เวลาตามนิยามที่ 1

$$t_{getmeta} = t_{request} + t_{metaservice} + t_{reply}$$

โดยที่ $t_{getmeta}$ เป็นเวลารวมที่ใช้ในการติดต่อกับโหนดบริหารในแต่ละครั้งเพื่อแจ้งถึงติดต่อใช้งานระบบพีวีเอฟเอส จากนิยามข้างต้นจะเป็นผลรวมของ $t_{request}$ เวลาในการร้องขอซึ่งต้องวิ่งผ่านระบบเครือข่ายจากลูกข่ายไปยังโหนดบริหารบวกกับ $t_{metaservice}$ เวลาที่โหนดบริหารใช้ในการให้บริการและ t_{reply} เวลาส่งผลติดต่อกลับไปยังลูกข่าย ซึ่งระยะเวลาในการติดต่อและให้บริการจะขึ้นอยู่กับอัตราความเร็วของระบบเครือข่าย ความหนาแน่นของเครือข่ายในขณะนั้น และปริมาณของผู้ขอใช้บริการ ซึ่งโครงการนี้จะใช้วิธีการสุ่มค่าเวลาในการให้บริการ เนื่องจากเวลาในส่วนของการติดต่อเพื่อร้องขอเมทาดาต้าเป็นเวลาด่วนน้อยของระบบ ในงานชิ้นนี้จึงได้กำหนดค่าตายตัวไว้ในโปรแกรม โดยใช้การกระจายแบบเอ็กซ์โปเนนเชียล

ในการอ่านหรือเขียนข้อมูลเมื่อลูกข่ายกระทำการเสร็จแล้วก็ติดต่อกลับมายังโหนดบริหารอีกครั้งเพื่อแจ้งให้ดำเนินการปิดไฟล์

3.1.2 ชั้นถ่ายโอนข้อมูล

เมื่อลูกข่ายได้รับข้อมูลเกี่ยวกับไฟล์ที่ต้องการแล้วลูกข่ายจะติดต่อไปยังไอโอโหนดเพื่อทำการอ่านหรือเขียนตามที่ต้องการ โดยจะใช้เวลาในการถ่ายโอนข้อมูลตามแต่สถาปัตยกรรมของระบบเครือข่าย บวกด้วยค่าหน่วยเวลาโดยรวม ดังแสดงได้ตามนิยามที่ 2

$$t_{transfer} = \frac{ds}{B} + t_{totaldelay}$$

โดยที่ $t_{transfer}$ เป็นระยะเวลาในการถ่ายโอนข้อมูลจากเครื่องลูกข่ายไปยังไอโอโหนดหรือจากไอโอโหนดมายังเครื่องลูกข่ายคำนวณได้จากการนำค่า ds ซึ่งเป็นค่าขนาดข้อมูลหารด้วยค่าอัตราการส่งผ่านข้อมูลหรือ B (Bandwidth) บวกกับค่า $t_{totaldelay}$ หรือ ค่าหน่วยเวลาที่ต้องเสียไปในระบบเครือข่าย แต่ในครั้งนี้จะใช้วิธีการสุ่มค่าจากการวัดจริง ผ่านฟังก์ชันการกระจายแบบ

เอ็กซ์โปเนนเชียล ซึ่งจะให้ค่าในช่วงใกล้เคียงกับค่าที่ส่งให้ฟังก์ชัน ทำให้ระบบสามารถจำลองเหตุการณ์ได้ใกล้เคียงของจริง

3.1.3 ชั้นอ่านเขียนข้อมูล

ในการอ่านหรือเขียนข้อมูลของไอโอโหนดจะกระทำการอ่านเขียนข้อมูลกับระบบไฟล์ปรกติของระบบ ซึ่งจะใช้เวลาในการอ่านเขียนข้อมูลตามแต่ความเร็วของระบบดิสก์ที่ใช้งานรวมกับกระบวนการในการเตรียมความพร้อมก่อนที่จะอ่านเขียนไฟล์ ตามนิยามที่ 3

$$t_{disk} = \frac{ds}{S} + t_{overhead}$$

โดยที่ t_{disk} คือเวลาในการเขียนหรืออ่านข้อมูล คำนวณได้จากการนำ ds ขนาดของข้อมูลหารด้วย S อัตราการให้บริการของดิสก์ บวกด้วย $t_{overhead}$ เวลาเวลาในการเข้าถึงข้อมูล เช่น ค่า seek time เป็นต้น โดยในโครงการนี้จะรับค่าอัตราการให้บริการของดิสก์จากผู้ใช้งาน ฟังก์ชันของกระจายแบบเอ็กซ์โปเนนเชียล เพื่อสุ่มค่าเวลาในการให้บริการออกมา เนื่องด้วยเวลาในส่วนของค่า $t_{overhead}$ นั้นใช้เวลาสั้น และทำการวัดเก็บค่าได้ยากในโครงการนี้จึงไม่ได้นำค่าดังกล่าวมาคำนวณ

3.2 แบบจำลองของพีวีเอฟเอส

จากตัวแบบข้างต้นสามารถสรุปเวลาในการให้บริการของระบบพีวีเอฟเอสได้ โดยการนำค่าเวลาในขั้นตอนต่างๆ มารวมกัน ดังแสดงไว้ในนิยามที่ 4

$$t_{service} = t_{getmeta} + t_{spread} + t_{transfer} + t_{disk}$$

สำหรับค่า t_{spread} เป็นค่าเวลาที่ใช้ในการแบ่งไฟล์ข้อมูลออกเป็นชิ้นเล็กๆ โดยในโครงการนี้ได้กำหนดให้ค่าเวลาในการแบ่งข้อมูลเป็นข้อมูลคลงที่สัมพันธ์กับจำนวนแพ็กเกจข้อมูลที่ถูกรวบรวมออกมา

ค่าที่ใช้ในการวัดและประเมินประสิทธิภาพของระบบไฟล์นิยมแสดงในรูปแบบของอัตราการถ่ายโอนข้อมูลต่อหน่วยเวลา เช่น สามารถอ่านข้อมูลได้ในอัตรา 20 เมกะไบต์ต่อวินาที เป็นต้น แต่ทั้งนี้ระบบพีวีเอฟเอส มีลักษณะการทำงานแบบขนานทำงานบนระบบคลัสเตอร์ซึ่งมีการอ่านเขียนไฟล์จากเครื่องคอมพิวเตอร์หลายเครื่องด้วยกัน ฉะนั้นค่าที่ใช้ในการวัดและประเมิน

ประสิทธิภาพของระบบนั้นจะนิยมใช้ค่าผลรวมของอัตราการถ่ายโอนข้อมูล (Aggregate Bandwidth) คำนวณได้โดยการนำค่าอัตราการถ่ายโอนข้อมูลของทุกๆ เครื่องในระบบมารวมกัน ดังแสดงได้ตามนิยามที่ 5

$$B_{agg} = \sum_{i=1}^{cp_n} \frac{ds_i}{t_i}$$

โดยที่ B_{agg} คือค่าแบนด์วิดท์โดยรวม (Aggregate Bandwidth) คำนวณได้จากผลรวมของแบนด์วิดท์ในการถ่ายโอนข้อมูลของโหนดคำนวณทุกโหนดมารวมกัน ในนิยามข้างต้นกำหนดให้ค่า cp_n แทนจำนวนโหนดคำนวณหรือเครื่องลูกข่ายที่เข้ามาใช้ระบบพีวีเอฟเอส ds แทนขนาดของข้อมูลที่โหนดคำนวณแต่ละเครื่องถ่ายโอนกับระบบพีวีเอฟเอส t แทนระยะเวลาที่ใช้ในการโอนถ่ายข้อมูลระหว่างโหนดคำนวณแต่ละเครื่องกับระบบพีวีเอฟเอส

3.3 ตัวแบบพีวีเอฟเอส

เริ่มจากการสร้างสถานะจำลองของระบบพีวีเอฟเอส มีการกำหนดคุณลักษณะต่างๆ ของระบบตามค่าที่รับมาจากผู้ใช้จากนั้นเข้าสู่กระบวนการของแถวคอย คือเริ่มมีการจ่ายงานไปยังเหตุการณ์ต่างๆ ตามลำดับที่เกิดขึ้น โดยเริ่มนับเวลาการจำลอง ลำดับเหตุการณ์จะไปตามลำดับดังต่อไปนี้

การคิวรีเมทอะดาต้า ก็จะจัดลำดับงานตามเหตุการณ์เวลาที่ใช้ในการบริการจะได้จากการสุ่มค่ารูปแบบการกระจาย (Distribution) ที่กำหนด นอกจากนั้นยังมีการกำหนดสถานะของหน่วยให้บริการด้วยว่าขณะนั้นหน่วยให้บริการนั้นว่างอยู่หรือไม่ ถ้าว่างอยู่ก็จะสามารถให้บริการได้ ถ้าไม่ว่างก็จะเกิดการรอซึ่งมีจำนวนของงานที่รอ หรือที่เรียกว่าแถวคอยเกิดขึ้น เมื่อประมาณผู้ใช้งานขึ้น จำนวนผู้ที่รออยู่ในแถวคอยก็มากขึ้นมากขึ้นตาม ส่งผลให้เวลาในการให้บริการโดยรวมมากขึ้น เมื่อนำเวลากับประมาณงานที่ให้บริการได้มาคำนวณเป็นอัตราส่วนระหว่างกัน ย่อมทำให้งานที่มีแถวคอยจำนวนมาก ให้ผลลัพธ์ที่ต่ำกว่าสายงานที่มีแถวคอยน้อย แต่ในด้านการใช้ประโยชน์จากระบบอย่างคุ้มค่าอาจให้ผลต่างกัน

แถวคอยในระบบพีวีเอฟเอส

- แถวคอยในการคิวรีเมทอะดาต้า จะกระทำเมื่อเริ่มต้นที่อ่านไฟล์ และจะเกิดขึ้นอีกครั้งเมื่อสิ้นสุดการใช้งานไฟล์นั้น การติดต่อใช้เวลาสั้นๆ เพื่อแจ้งถึงการขอใช้ไฟล์เท่านั้น จึงใช้เวลาไม่มาก
- แถวคอยในการอ่านเขียนดิสก์ ระบบดิสก์ก็สามารถให้บริการงานได้ในปริมาณที่

เอกสารนี้เป็นเอกสารจำกัดในหนึ่งช่วงเวลาเช่นกัน การเราจำเป็นต้องทราบก่อนว่าหน่วยให้บริการดิสก์นี้ราคาไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้บริการได้เป็นอัตราเท่าไรต่อวินาที ในงานชิ้นนี้ใช้หน่วยเป็นเมกะไบต์ต่อวินาที เช่น 50 MB/Sec เป็นต้น ตรงจุดนี้เราจะใช้การสุ่มค่า ที่อยู่ในช่วงที่สอดคล้องกับความ เป็นจริง เพื่อประมาณการว่าจะต้องใช้เวลามากหรือน้อยกับงานในแต่ละชิ้น

- แถวคอยในการส่งผ่านระบบเครือข่าย ระบบเครือข่ายเป็นอีกจุดสำคัญของระบบ เพราะระบบคลัสเตอร์เชื่อมต่อกันผ่านระบบเครือข่าย โดยมากจะใช้สวิตซ์ เป็นหลัก

3.4 ขั้นตอนการดำเนินงาน

การศึกษาโครงการนี้จะดำเนินการศึกษาวิธีการทำงานของระบบไฟล์แบบขนานพีวีเอฟเอส และหลักการประเมินสมรรถนะทางคอมพิวเตอร์ด้วยวิธีการใช้การจำลองด้วยเทคนิคการใช้ตัวแบบแถวคอย และพัฒนาระบบสร้างเป็นโปรแกรมคอมพิวเตอร์ เพื่อประยุกต์ใช้การจำลองการทำงาน ของระบบไฟล์แบบขนานพีวีเอฟเอสบนระบบคลัสเตอร์ โดยใช้ไลบรารีเหตุการณ์ไม่ต่อเนื่องซึ่งเป็นภาษาซีพลัสพลัส(C++) ในตัวโปรแกรมจะแบ่งเป็นสองส่วนคือ ส่วนประมวลผล และส่วนแสดงผล ในส่วนประมวลผล โปรแกรมจะถูกพัฒนาขึ้นด้วยภาษาซีพลัสพลัส โดยใช้โปรแกรมไมโครซอฟท์วิซวลซีพลัสพลัส (VC++) โดยพัฒนาในรูปแบบบรรทัดคำสั่ง และในส่วนแสดงผลถูกพัฒนาโดยใช้โปรแกรมบอร์แลนด์เดลฟี (Delphi) โดยมีขั้นตอนในการดำเนินงาน ดังนี้

1. ศึกษาและทดลองใช้โปรแกรมพีวีเอฟเอส ในขั้นตอนนี้ได้ศึกษาค้นคว้าวิธีการทำงานแนวคิด ของระบบไฟล์แบบขนานพีวีเอฟเอส พร้อมทั้งได้ทดลองติดตั้งใช้งาน และปรับแต่งระบบในรูปแบบต่างๆ ได้วัดและเก็บผลการทดลองเบื้องต้นเพื่อศึกษาพฤติกรรมของระบบ
2. ออกแบบตัวแบบที่จะใช้ในการประเมินประสิทธิภาพของระบบไฟล์แบบขนานพีวีเอฟเอส โดยได้สรุปการข้อมูลที่ได้การศึกษาในขั้นต้น พบว่าปัญหาหลักของระบบจะอยู่การเกิดคอขวด ณ บริเวณสองจุดใหญ่ๆ คือ ที่ดิสก์หรือระบบเครือข่าย อันจะเป็นประเด็นหลักที่จะทำการศึกษา
3. วัดประสิทธิภาพของตัวแปรที่จะใช้ในการทดสอบ ได้ใช้โปรแกรมวัดประสิทธิภาพไปตรวจวัดความเร็วในการสื่อสารข้อมูลของระบบเครือข่ายที่ใช้เชื่อมต่อกลัสเตอร์ และได้วัดประสิทธิภาพความเร็วในการอ่านเขียนข้อมูลของฮาร์ดดิสก์ เพื่อใช้ค่าที่วัดจริงในการจำลอง ซึ่งน่าจะให้ผลได้ใกล้เคียงกับความเป็นจริงมากที่สุด
4. นำข้อมูลที่ได้จากข้อ 3. มาป้อนเป็นอินพุตให้กับ โปรแกรมที่ได้สร้างไว้

5. ปรับเปลี่ยนพารามิเตอร์บางตัวเพื่อเปรียบเทียบผลลัพธ์ได้ เพื่อหาตำแหน่งที่ให้ผลลัพธ์ที่ตรงกับความต้องการ

3.5 การออกแบบโปรแกรมจำลอง

กระบวนการออกแบบและการทำงานของ โปรแกรมประยุกต์ มีดังนี้

3.5.1 การออกแบบโปรแกรม

การออกแบบ โปรแกรมประยุกต์นี้จะมีลักษณะเป็นโปรแกรมเชิงวัตถุ (Object Oriented Program) โดยจัดแบ่งหน้าที่การทำงานส่วนต่างๆ เป็น โมดูล (Module) ดังนี้

- ส่วนการรับอินพุตจากผู้ใช้
- ส่วนการจำลองเหตุการณ์ด้วยวิธีการแบบเครือข่ายแถวคอย
ในส่วนจะมีการสร้างสถานะจำลองของระบบไฟล์แบบขนานพีวีเอฟเอสบนระบบคลัสเตอร์โดยมีจำนวนโหนดตามที่ได้รับค่ามาจากผู้ใช้ จากนั้นจะเริ่มการจำลองพร้อมการเก็บค่าทางสถิติที่สำคัญ ใช้การกระจายแบบ Exponential และ Poisson
- ส่วนการแสดงผลออกมารูปแบบตัวเลข
- ส่วนการแสดงผลด้วยกราฟ
- ส่วนบันทึกข้อมูลในรูปแบบคั่นด้วยลูกน้ำ (CSV Format)

3.5.2 การทำงานของโปรแกรมประยุกต์

การทำงานของ โปรแกรมประยุกต์แบ่งเป็นส่วนต่างๆ ดังนี้

- ส่วนของการทำงานในการจำลองสภาพการทำงานของคลัสเตอร์ หลังจากรับอินพุตจากผู้ใช้ โปรแกรมจะส่งผ่านไปยังตัวโปรแกรมหลักที่เป็นแบบคอมมานด์ไลน์ เพื่อประมวลผล จำลองสถานการณ์ตามข้อมูลที่ได้รับมา
- ส่วนของการแสดงผล เป็นส่วนที่รับค่าจากโปรแกรมคอมมานด์ไลน์ เพื่อนำผลลัพธ์มาแสดงแก่ผู้ใช้ แบ่งเป็น 2 ส่วน คือ
 - ส่วนแสดงผลในรูปแบบตัวเลข ซึ่งแสดงอยู่ในตาราง
 - ส่วนแสดงผลเป็นกราฟ
- ส่วนการบันทึกผลลัพธ์ ออกเป็นไฟล์ เพื่อให้สามารถเก็บค่า และนำไปใช้ร่วมกับโปรแกรมอื่นๆ ได้

บทที่ 4

ผลการทดลอง

จากการทดลอง สามารถสรุปผลการทดลองการทำงานจากการพัฒนาโปรแกรมแบบจำลองด้วยเทคนิคตัวแบบแถวคอก แบบเครือข่ายแถวคอก โดยแบ่งการอธิบายออกเป็นส่วนต่างๆ ดังนี้

1. ส่วนการเตรียมข้อมูล
2. ส่วนการทำงานของโปรแกรมประยุกต์
3. ส่วนการแสดงผลลัพธ์

4.1 ระบบที่พัฒนา

ระบบส่วนแรกเป็นแกนหลักของซอฟต์แวร์คือส่วนที่ทำหน้าที่ในการจำลองสถานการณ์พัฒนาขึ้นด้วยภาษาซีพลัสพลัส เป็นแบบบรรทัดคำสั่ง (Command Line) สามารถรับอินพุตพารามิเตอร์จากผู้ใช้ โดยผู้ใช้สามารถปรับพารามิเตอร์ได้ครอบคลุมที่ส่วนที่สำคัญต่อการจำลอง รูปที่ 5.1 แสดงภาพของคำสั่งและพารามิเตอร์ต่างๆ ที่ผู้ใช้สามารถเรียกใช้และปรับแต่งได้ โดยมีพารามิเตอร์หลักๆ ที่สำคัญดังต่อไปนี้

- จำนวนโหนดประมวลผล ซึ่งหมายถึงจำนวนเครื่องลูกข่ายที่จะติดต่อเข้าใช้งานระบบ
- จำนวนไอโอโหนด หมายถึง จำนวนเครื่องที่ทำหน้าที่ให้บริการไฟล์ในระบบพีวีเอฟเอส
- อัตราการรับส่งข้อมูล (Bandwidth) มีหน่วยเป็นเมกะไบต์ (MB)
- ขนาดของไฟล์ข้อมูล (File Size) มีหน่วยเป็นเมกะไบต์ (MB)
- ขนาดที่ตัดแบ่ง (Striped Size) มีหน่วยเป็นกิโลไบต์ (KB)
- อัตราการเขียนข้อมูล (Write Speed) มีหน่วยเป็นเมกะบิตต่อวินาที (Mb/sec)
- อัตราการอ่านข้อมูล (Read Speed) มีหน่วยเป็นเมกะบิตต่อวินาที (Mb/sec)

```

C:\WINDOWS\system32\cmd.exe
operable program or batch file.

D:\my_doc\hpcnc\hypersim\models\simmod\Debug>simmod
pufs Simulation for finding the performance on concurrent read and write
Usage: pufssim [options] trace
-d, --debug=[INT]      debug level
-s, --seed=[INT]      random seed
-t, --trace            enable trace
--netlogger           enable netlogger trace
-r, --refnum=[INT]    number of references
-c, --cpNode=[INT]    Amount of Compute nodes, Default 1 Compute node
-i, --ioNode=[INT]    Amount of I/O nodes
-b, --bandwidth=[DOUBLE] Network speeds (Core switch Mbps)
-z, --stripesize=[INT] Stripe size (KB)
-f, --filesize=[DOUBLE] File size (MB)
-j, --writeSpeed=[DOUBLE] Compute speed (MB/sec)
-k, --readSpeed=[DOUBLE] I/O speed (MB/sec)
-h, --help            show this help

Example:
pufssim -m 1048576 -c 65536 -b 8192 "u(0,1048576)"
pufssim "simpletrace(trace.in)"
pufssim "e(10240)"

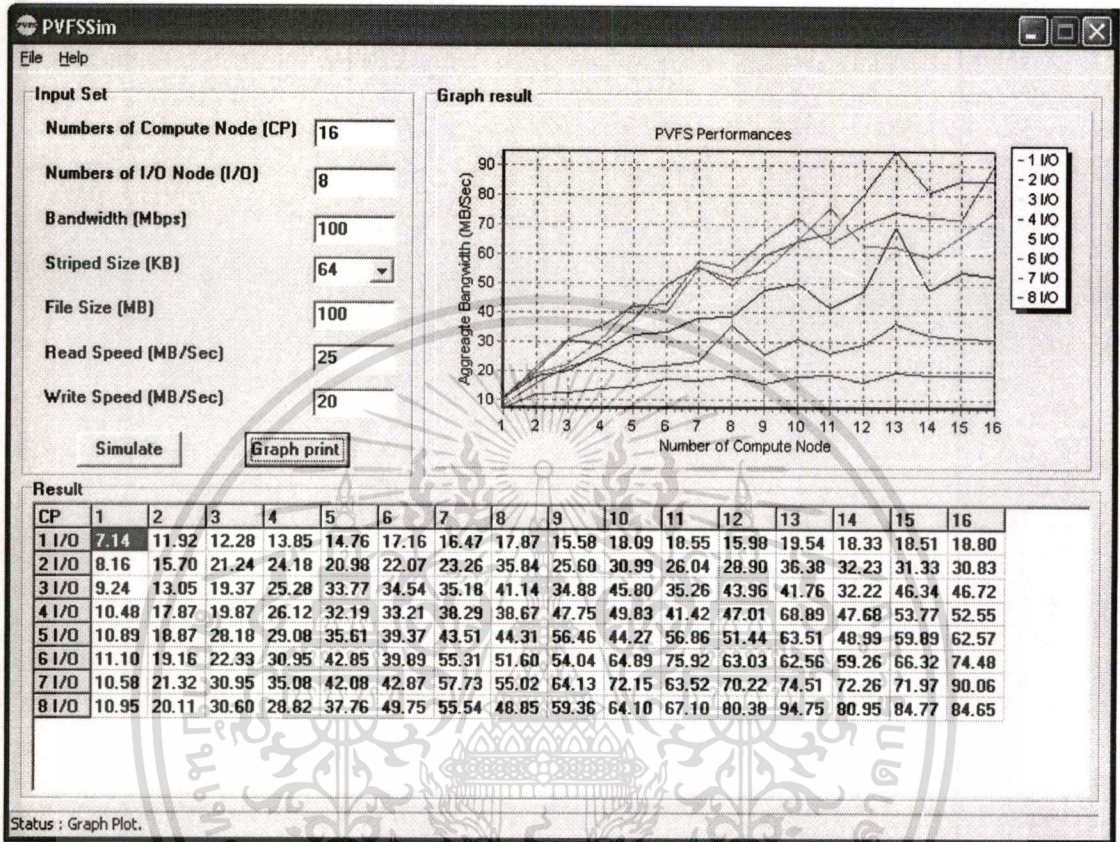
D:\my_doc\hpcnc\hypersim\models\simmod\Debug>

```

รูปที่ 4.1 ตัวอย่างซอฟต์แวร์แบบบรรทัดคำสั่ง

ในส่วนของการพัฒนาตัวโปรแกรมหลักในส่วนนี้ พัฒนาโดยภาษาซีพลัสพลัส โดยใช้ไลบรารีเหตุการณ์ไม่ต่อเนื่อง ไฮเปอร์ซิม ดังที่ได้กล่าวไว้ในบทก่อนหน้า โดยมุ่งสนใจในส่วนของการวิเคราะห์จากแถวคอยของระบบเครือข่ายและโหนดที่ทำหน้าที่เป็นอินพุทเอาต์พุทเป็นหลัก งานที่พัฒนาข้างต้นตั้งอยู่บนพื้นฐานของการใช้งาน โปรแกรมแบบบรรทัดคำสั่ง ทำให้ยากต่อการใช้งาน ปัจจุบันอยู่ในขั้นการออกแบบส่วนติดต่อผู้ใช้ที่เป็นกราฟฟิก เพื่อที่จะให้สามารถรับพารามิเตอร์ต่างๆ ไปประมวลผลและแสดงผลทั้งในรูปแบบตัวเลขและกราฟในหน้าจอเดียวกัน โดยจะกล่าวถึงในหัวข้อถัดไป

4.2 ส่วนการทำงานของโปรแกรมประยุกต์



รูปที่ 4.2 ตัวอย่างหน้าจอของซอฟต์แวร์ PVFSSim

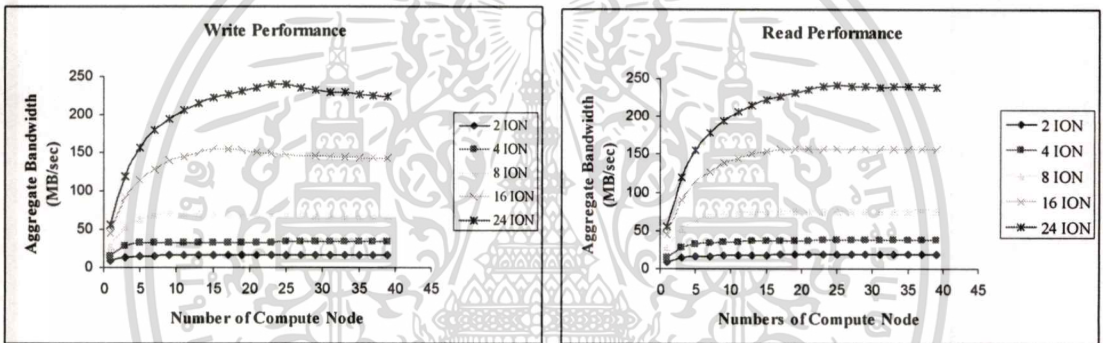
รูปที่ 4.2 แสดงตัวอย่างหน้าจอของซอฟต์แวร์ สามารถแบ่งมีหน้าที่การทำงานหลักๆ ดังต่อไปนี้ ส่วนรับค่าจากผู้ใช้

1. ส่วนรับข้อมูลจากผู้ใช้ ซึ่งจะรับข้อมูลต่างๆ ที่ต้องใช้ในการจำลอง เพื่อส่งให้โปรแกรมที่เป็นแบบคอมมานด์ไลน์ รับไปทำงานต่อ
2. ส่วนแสดงผล แบ่งออกได้เป็นสองส่วนย่อย ดังต่อไปนี้
 - ส่วนแสดงผลในรูปแบบตัวอักษร เป็นส่วนที่แสดงผลจากการจำลองในรูปแบบตัวเลข
 - ส่วนแสดงผลในรูปแบบกราฟ เป็นการแสดงผลในรูปแบบกราฟ โดยสร้างจากผลลัพธ์ที่ได้มาจากโปรแกรม

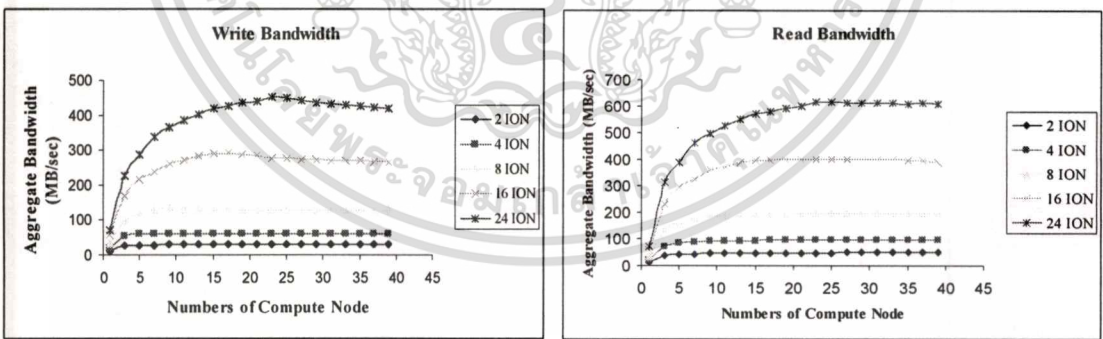
3. การนำออกข้อมูลในรูปแบบไฟล์ ผลลัพธ์ที่ได้จะสามารถนำออกในรูปแบบของไฟล์ข้อมูลที่ค้นด้วยเครื่องหมายลูกน้ำ (CSV) ซึ่งข้อมูลดังกล่าวสามารถนำไปใช้ประโยชน์ต่างๆต่อไปได้ เช่น การนำไปแสดงผลผ่านโปรแกรมเอ็กเซลล์ (Excel)

4.3 ผลการทดลอง

Moungmingsuk and Uthayopas. (2004) ตามรูปที่ 4.2 และ 4.3 แสดงผลการทดลองจำลองการใช้งานระบบไฟล์แบบขนานพีวีเอฟเอส โดยจำลองสถานการณ์เปรียบเทียบกับอัตราการรับส่งข้อมูลของระบบเครือข่ายต่างกัน ระหว่างฟาสต์อีเทอร์เน็ตและกิกะบิตอีเทอร์เน็ต โดยให้จำนวนไอโหนดและโหนดประมวลผลในจำนวนที่ต่างกัน



รูปที่ 4.3 กราฟแสดงประสิทธิภาพของระบบพีวีเอฟเอส บนเครือข่ายแบบฟาสต์อีเทอร์เน็ต



รูปที่ 4.4 กราฟแสดงประสิทธิภาพของระบบพีวีเอฟเอส บนเครือข่ายแบบกิกะบิตอีเทอร์เน็ต

การทดลองข้างต้น ใช้อัตราการอ่านข้อมูลของดิสก์ที่ 22 MB/sec และการเขียนข้อมูลที 15 MB/sec ใช้ข้อมูลขนาด 100 MB ขนาดในการตัดแบ่ง 16 KB โดยกำหนดให้ทุกๆ โหนดประมวลผลเรียกใช้ข้อมูลพร้อมๆ กัน โดยสมมุติให้ขนาดความยาวของแถวค้อยในเครือข่ายและในดิสก์มีไม่จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการทดลองในขั้นต้นทำให้ทราบว่าจำนวนไอโอโหนดส่งผลต่อประสิทธิภาพของการให้บริการโดยรวม ซึ่งปริมาณของเครื่องควมสัมพันธ์ความเร็วของระบบเครือข่าย บนระบบเครือข่ายแบบฟาสต์อีเทอร์เน็ตเปรียบเทียบกับกิกะบิตอีเทอร์เน็ตแสดงให้เห็นอย่างชัดเจนว่าเกิดคอขวดขึ้นที่ระบบเครือข่าย จากกราฟจะเห็นว่าประสิทธิภาพของระบบจะเพิ่มขึ้นบนระบบเครือข่ายแบบกิกะบิต แต่ยังคงมีพฤติกรรมในการหยุดชะงักของกราฟ ณ จุดที่ปริมาณโหนดประมวลผลและไอโอโหนดมีปริมาณใกล้เคียงกัน ผลการทดลองนี้ให้ผลสอดคล้องกับการวัดจากระบบจริงของ Cams et. at. (2004).



บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผลโครงการ

โครงการพัฒนาระบบประเมินประสิทธิภาพระบบไฟล์แบบขนานพีวีเอฟเอส บนระบบพีซีคลัสเตอร์ นี้มุ่งวิเคราะห์ระบบไฟล์แบบขนานโดยเน้นในส่วนของระบบเครือข่ายและระบบอินพุทเอาต์พุทเป็นหลัก โปรแกรมสามารถจำลองสถานการณ์ของระบบคลัสเตอร์ที่ใช้จำนวนเครื่อง ความเร็ว ปริมาณของข้อมูล ที่แตกต่างกันได้ ซึ่งการจำลองจะเป็นส่วนหนึ่งที่จะช่วยให้ผู้บริหารระบบคลัสเตอร์สามารถปรับแต่งเพื่อให้สามารถใช้ประสิทธิภาพของระบบที่สูงที่สุด ซึ่งจะเป็นการคุ้มค่าต่อการลงทุน และยังสามารถใช้ประกอบงานวิจัย โดยสามารถใช้ค่าจากการจำลองเพื่อสนับสนุน ผลการประเมินประสิทธิภาพของระบบด้วยการวัดค่าจากระบบจริง ซึ่งตามทฤษฎีของการประเมินประสิทธิภาพทางคอมพิวเตอร์นั้นการที่จะมั่นใจได้ว่าผลที่ได้นั้นจะต้องจำเป็นต้องมีข้อมูลอื่นๆ สนับสนุน

ในทางกลับกัน ถ้าเราสามารถปรับแต่งค่าตัวแปรในการจำลองจนได้ผลสอดคล้องกับผลที่วัดได้จากระบบจริง ฉะนั้นน่าจะอนุมานได้ว่าการจำลองของเรานั้นมีความเที่ยงตรง ถ้าเราเอาตัวแบบการจำลองนั้น ไปจำลองระบบในสถานะการณ์อื่น ก็น่าจะได้ผลลัพธ์ที่ถูกต้องและเป็นไปตามระบบจริง

5.2 ข้อเสนอแนะ

การพัฒนาโปรแกรมในส่วนนี้ ยังคงมีตัวแปรบางตัวที่ไม่ได้ถูกหยิบยกขึ้นมาพิจารณา และค่าตัวแปรบางค่า ผู้ใช้โปรแกรมยังไม่อาจกำหนดได้ ทำให้ผลที่ได้อาจไม่ละเอียดมาก ถ้ามีการพิจารณาที่ส่วนอื่นๆ เพิ่มขึ้นย่อมจะทำให้โปรแกรมสามารถประเมินผล ได้ใกล้เคียงความเป็นจริงมากยิ่งขึ้น

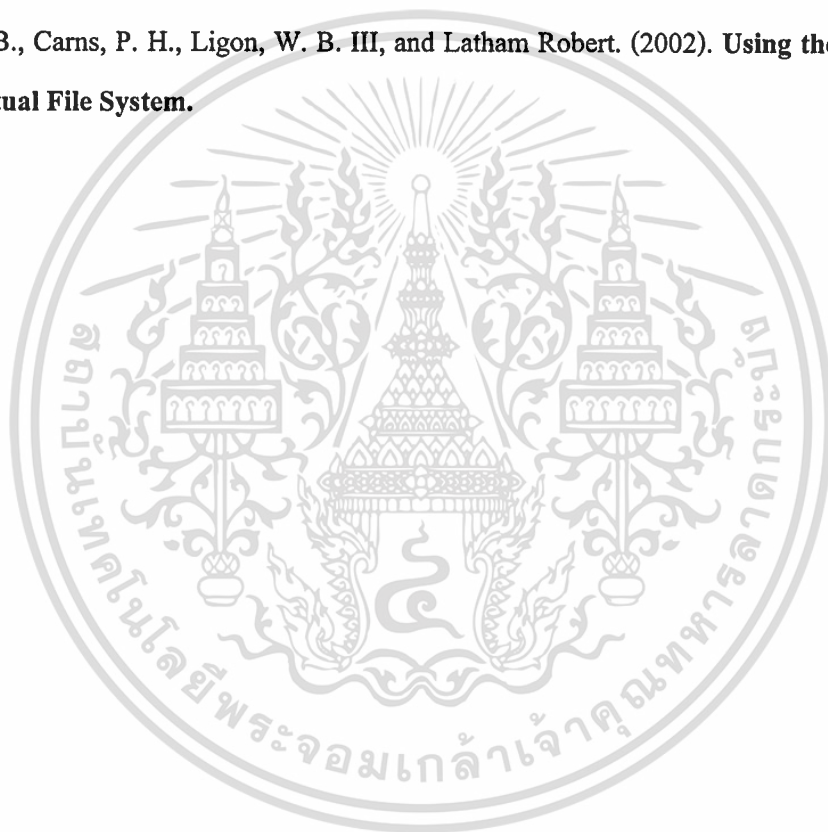
บรรณานุกรม

- ฤดี มาสุจันท์. 2546. การวิจัยดำเนินงาน. กรุงเทพฯ : แผนกตำรา คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- วรา วราวิทย์. 2545. การประเมินประสิทธิภาพ ของระบบคอมพิวเตอร์. ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ. [Online]. Available: <http://hpc.ee.kmitnb.ac.th/~vara/perf/perf.html>
- Avery Ching, Choudhary, A., Wei-keng Liao, Ross, R., Gropp, W. (2002) "Noncontiguous I/O through PVFS." **IEEE International Conference on Cluster Computing (CLUSTER'02)**. September 23 - 26, Chicago, Illinois
- Barro, J., Tourino, J., Doallo, R., Gulias, V. (2002). "Performance Modeling and Evaluation of MPI-I/O on a Cluster." **Journal of Information Science and Engineering**, September, Vol.18 No.5, pp.825-836.
- Carns, P. H., Ligon, W. B. III, Ross, R. B., and Thakur, R. (2000). "PVFS: A Parallel File System for Linux Clusters." **In Proceedings of the 4th Annual Linux Showcase and Conference**, pages 317-327, Atlanta, GA, USENIX Association
- Feng D., Jiang H., Zhu Y., (2003). "I/O Response Time In A Fault-Tolerant Parallel Virtual File System." **Technical Report. Department of Computer Science and Engineering**, University of Nebraska-Lincoln. July.
- Ligon, W.B., III Ross, R.B. (1996). "Implementation and Performance of a Parallel File System for High Performance Distributed Applications." **High Performance Distributed Computing, 5th IEEE International Symposium**. IEEE. USA
- May, M. John. (2001). **Parallel I/O for High Performance Computing**. Academic Press United States of America (p. 74-75).
- Milfeld Kent, Purkayastha Avijit, Guiang Chona. (2002). "Experiences with the Parallel Virtual File System (PVFS) in Linux Clusters." **Accepted for the 3rd LCI International Conference on Linux Clusters**. at St. Petersburg, October, Florida.

Moungmingsuk Kittirak, Uthayopas Putchong. "Performance Modeling of PVFS Parallel File System on Commodities PC Cluster." **In the Proceeding of The 8th Annual National Symposium on Computational Science and Engineering**, Nakhon Ratchasima, Thailand, 21-23 July, 2004.

Phatanapherom Sugree, Uthayopas Putchong, Kachitvichyanukul Voratas. (2003). "Fast Simulation Model for Grid Scheduling using HyperSim." **Winter Simulation Conference**. New Orleans, Louisiana, USA. December 7-10.

Ross, R. B., Carns, P. H., Ligon, W. B. III, and Latham Robert. (2002). **Using the Parallel Virtual File System.**



ประวัติผู้เขียน

ชื่อ-นามสกุล

ว่าที่ร้อยตรีกิตติรักษ์ ม่วงมิ่งสุข

วัน เดือน ปีเกิด

5 กันยายน 2522

ที่อยู่

91 ริมคลองซึกพระ แขวงบางขุนนนท์ เขตบางกอกน้อย กรุงเทพฯ 10700

ประวัติการศึกษา

พ.ศ. 2543 บริหารธุรกิจบัณฑิต สาขาระบบสารสนเทศ สถาบันเทคโนโลยีราชมงคล

พ.ศ. 2545 ศิลปศาสตรบัณฑิต สาขารัฐศาสตร์(บริหารรัฐกิจ) มหาวิทยาลัยรามคำแหง

พ.ศ. 2546 ประกาศนียบัตรบัณฑิต สาขาวิชาชีพครู มหาวิทยาลัยรามคำแหง

ความชำนาญเฉพาะด้าน

1. ระบบฐานข้อมูลบนเว็บ
2. ระบบคอมพิวเตอร์สมรรถนะสูง

ประสบการณ์การทำงาน

พ.ศ.2545-2546 เจ้าหน้าที่พัฒนาระบบ บริษัทภาคใต้เชื้อเพลิง จำกัด (มหาชน)

ปัจจุบัน ผู้ช่วยวิจัย ศูนย์วิจัยคอมพิวเตอร์สมรรถนะสูงและเครือข่ายคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์