รายงานการวิจัย

เรื่อง

การประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์บนระบบกริด

A Distributed Rendering Method for Visualization of Medical Dataset on a Grid Environment

ดร.ศิริลักษณ์ หล่อพันธ์มณี

มหาวิทยาลัยสวนดุสิต

2560

ลิขสิทธิ์ของมหาวิทยาลัยสวนดุสิต

รายงานการวิจัย

เรื่อง

การประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์บนระบบกริด

A Distributed Rendering Method for Visualization of Medical Dataset on a Grid Environment

ดร.ศิริลักษณ์ หล่อพันธ์มณี

มหาวิทยาลัยสวนดุสิต

2560

ลิขสิทธิ์ของมหาวิทยาลัยสวนดุสิต

(งานวิจัยนี้ได้รับทุนอุดหนุนจากงบประมาณแผ่นดินด้านการวิจัย ปีงบประมาณ 2558)

หัวข้อวิจัย การประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์บนระบบกริด

ผู้ดำเนินการวิจัย ดร.ศิริลักษณ์ หล่อพันธ์มณี

หน่วยงาน หลักสูตรวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี
มหาวิทยาลัยสวนดุสิต

ปี พ.ศ. 2560

การวิจัยครั้งนี้มีวัตถุประสงค์ ดังนี้

1. ได้โปรแกรมต้นแบบในการประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์ในระบบกริด
2. เพื่อลดระยะเวลาของการประมวลผลภาพทางการแพทย์ที่มีจำนวนมากและขนาดใหญ่ได้อย่างมีประสิทธิภาพ
3. เพื่อให้สามารถใช้ทรัพยากรที่มีอยู่ในระบบได้อย่างมีประสิทธิภาพ
4. เพื่อศึกษาผลของการประมวลผลภาพแบบกระจายในระบบกริดกับระบบเดิม

ในการทดลองได้ใช้ Data Set เป็น 3D Dental Scans ประกอบด้วย 167 ภาพย่อย (slices) ที่มีความละเอียด 256 x 256 x 256 พิกเซล (pixel) และมีขนาดไฟล์ทั้ง 58 MB เครื่องมือที่ใช้ในการวิจัยมีดังนี้

คุณลักษณะของ Software

1. CentOS version 6.5
2. Globus Toolkit version 6
3. VTK Toolkit Library version 5.8
4. Ganglia 3.5.12
5. Java version 1.8

คุณลักษณะของ Hardware

1. เครื่องคอมพิวเตอร์ลูกข่าย 9 เครื่อง ประกอบด้วย
   - จำนวน CPU ระหว่าง 1 – 3 cpus
   - ความเร็วของ CPU 2.4 GHz
   - หน่วยความจำ (Ram) 2 GB
2. เครื่องคอมพิวเตอร์แม่ข่าย 1 เครื่อง ประกอบด้วย
   - จำนวน CPU ระหว่าง 4 cpus
   - ความเร็วของ CPU 2.4 GHz
   - หน่วยความจำ (Ram) 2 GB

3. ระบบเครือข่าย เครื่องคอมพิวเตอร์ทุกเครื่องเชื่อมต่อด้วยสาย LAN ความเร็ว 1000 MB Ethernet

ผลการวิจัยมีดังนี้ การเปรียบเทียบระหว่างโปรแกรมระบบที่พัฒนาขึ้น (Auto) และโปรแกรมพื้นฐาน (Manual) พบว่า โปรแกรมที่พัฒนาขึ้น สามารถประมวลผลภาพทางการแพทย์ได้อย่างมีประสิทธิภาพในด้านของเวลาที่ประมวลผลและสามารถกระจายการประมวลผลไปยังเครื่องคอมพิวเตอร์ลูกข่ายอย่างมีประสิทธิภาพในด้านของเวลาเช่นกัน อย่างไรก็ตามเมื่อเปรียบเทียบในด้านของการโอนย้ายข้อมูลระหว่างเครื่องคอมพิวเตอร์ในระบบที่ออกแบบมา พบว่า เมื่อจำนวนเครื่องคอมพิวเตอร์มีจำนวนเพิ่มขึ้น การโอนย้ายข้อมูลยังทำได้ไม่ดีพอ ทั้งนี้เพราะในการออกแบบโปรแกรม นักวิจัยเน้นในการประมวลผลที่เครื่องคอมพิวเตอร์แต่ละเครื่องให้มีประสิทธิภาพ ดังนั้นจึงทำการแบ่งภาพออกตามจำนวน CPU ในการประมวลผล เพื่อให้ใช้ทรัพยากรอย่างมีประสิทธิภาพและการประมวลผลที่รวดเร็วยิ่งขึ้น ผลที่ตามมาคือมีจำนวนภาพ Render เกิดขึ้นเป็นจำนวนมากซึ่งมีผลทำให้ภาพเหล่านั้นต้องเข้าคิวรอในระบบเครือข่ายของระบบเป็นผลทำให้ระยะเวลาในการโอนย้ายข้อมูลทำได้ไม่ดีพอเมื่อเทียบกับโปรแกรมพื้นฐาน (Manual)

Research Title	A Distributed Rendering Method for Visualization of Medical Datasets on a Grid Environment
Researcher	Dr.Siriluck Lorpunmanee
Organization	Faculty of Science and Technology Suan Dusit University
Year	2017

This research had the following purposes:

1. To obtain the prototype program of processing medical information distribution in a grid system
2. To effectively reduce the processing time for medical pictures in large quantity and size
3. To utilize system resources effectively
4. To study the processing of pictures distribution in a grid system and a normal system

The data set used in this study consisted of 3D dental scans containing 167 slices with resolution of 256 x 256 x 256 pixel. The total file size was 58 MB. The research tools were as follows:

Software specification

1. CentOS version 6.5
2. Globus Toolkit version 6
3. VTK Toolkit Library version 5.8
4. Ganglia 3.5.12
5. Java version 1.8

Hardware specification

1. Specification of 9 Client computers
   - CPU: 1-3 cpus
   - SPEED: 2.4 GHz
   - RAM: 2 GB
2. Specification of 1 Server
   - CPU: 4 cpus

- SPEED: 2.4 GHz
- RAM: 2 GB

3. Network – all computers were connected by LAN 1000 MB Ethernet

The result of the comparison between the automated and manual programs shows that the automated program was more effective in terms of medical pictures processing time. It also distributed the processing to its clients effectively in terms of time. However, the speed of data transfer among clients in the automated program decreased when the number of clients increased. This is the result of the program design from the researcher who emphasized processing in each client effectively by separating pictures according to the number of processing CPUs in order to utilize resources more effectively and the ability of faster processing. As a result, there were a lot of render pictures queued up in the network and delayed data transfer time compared to the manual program.

กิตติกรรมประกาศ

งานวิจัยฉบับนี้มีจุดมุ่งหมายเพื่อพัฒนาโปรแกรมต้นแบบในการประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์ในระบบกริดให้สามารถประมวลผลข้อมูลภาพที่มีจำนวนมากและขนาดใหญ่ได้อย่างมีประสิทธิภาพ ผู้วิจัยขอขอบคุณ สำนักงานคณะกรรมการวิจัยแห่งชาติ (วช.) ที่ได้สนับสนุนทุนการวิจัยครั้งนี้

การดำเนินการวิจัยมิอาจสำเร็จลุล่วงไปได้หากปราศจากความร่วมมือของคณาจารย์ในคณะวิทยาศาสตร์และเทคโนโลยีที่ให้การสนับสนุนการใช้อุปกรณ์ด้านฮาร์ดแวร์ และซอฟต์แวร์ต่าง ๆ ในการพัฒนาระบบ รวมถึงสถานที่ในการดำเนินการจัดทำวิจัย จนโครงการนี้สำเร็จลุล่วงไปด้วยดี

ท้ายนี้ผู้เขียนขอกราบขอบพระคุณบิดา มารดา ที่ให้การอุปการะอบรมเลี้ยงดู ตลอดจนส่งเสริมการศึกษา ตลอดจนนางฟ้าตัวน้อย ด.ญ.ชมพูนุท หล่อพันธ์มณี ที่ให้กำลังใจเป็นอย่างดี และภรรยา น.ส.ช่อผกา พันธุ์ภักดีวงษ์ ที่ค่อยให้กำลังใจตลอดเวลา อีกทั้งขอขอบคุณเพื่อน ๆ ที่ให้การสนับสนุนและช่วยเหลือด้วยดีเสมอมา และขอขอบพระคุณเจ้าของเอกสารและงานวิจัยทุกท่านที่ผู้ศึกษาค้นคว้าได้นำมาอ้างอิงในการทำวิจัย จนกระทั่งงานวิจัยฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี

ดร.ศิริลักษณ์ หล่อพันธ์มณี

2560

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ค
กิตติกรรมประกาศ	จ
สารบัญ	ฉ
สารบัญตาราง	ซ
สารบัญภาพ	ฌ
สารบัญแผนภูมิ	ญ
บทที่ 1 บทนำ	1
ความเป็นมาและความสำคัญ	1
วัตถุประสงค์ของการวิจัย	2
ขอบเขตการวิจัย	3
คำจำกัดความที่ใช้ในงานวิจัย	3
ประโยชน์ที่คาดว่าจะได้รับ	4
บทที่ 2 แนวคิด ทฤษฎี เอกสารและงานวิจัยที่เกี่ยวข้อง	5
มาตรฐานของการจัดการภาพทางแพทย์	5
เทคโนโลยีกริด (Grid Technology)	7
สถาปัตยกรรมของระบบกริด	8
โกลบัสทูลคิท (Globus Toolkit)	10
วีทีเคทูลคิท (VTK: Visualization Toolkit)	12
การสร้างภาพวิชวลไลเซชัน (Visualization)	14
งานวิจัยที่เกี่ยวข้องกับการประมวลผลภาพทางการแพทย์บนระบบกริด	14
กรอบแนวคิดในการวิจัย (Conceptual Framework)	18
สรุป	18

หน้า

บทที่ 3 วิธีดำเนินการวิจัย 19
รูปแบบการดำเนินการวิจัย (Research Operational Framework) 20
ประชากรและการสุ่มกลุ่มตัวอย่าง 32
เครื่องมือในการวิจัยและการตรวจสอบคุณภาพเครื่องมือ 32
การเก็บรวบรวมข้อมูล 32
การวิเคราะห์ข้อมูล 33
สรุป 33

บทที่ 4 ผลการวิจัย 34
การออกแบบการทดลอง (Testbed Implementation) 34
ผลการทดลองด้านระยะเวลาการประมวลผล 36
การทดลองระยะเวลาในการประมวลผลของแต่ละเครื่อง 38
การทดลองระยะเวลาในการโอนย้ายข้อมูล 40
สรุป 41

บทที่ 5 สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ 42
สรุปผลการวิจัย 42
อภิปรายผล 42
ข้อเสนอแนะในการนำผลการวิจัยไปใช้ 43
ข้อเสนอแนะในการทำวิจัยครั้งต่อไป 44

บรรณานุกรม 45
บรรณานุกรมภาษาไทย 45
บรรณานุกรมภาษาต่างประเทศ 45

ภาคผนวก 51
ภาคผนวก ก ขั้นตอนการติดตั้ง Cmake และ VTK Library 52
ภาคผนวก ข ขั้นตอนในการติดตั้ง Globus Toolkit 56

ประวัติผู้วิจัย 61

สารบัญตาราง

ตารางที่		หน้า
1.1	แสดงประโยชน์ที่ได้รับจากงานวิจัย	4
4.1	ชุดข้อมูลในการทอสอบ	34
4.2	คุณลักษณะคอมพิวเตอร์	34
4.3	ระยะเวลาในการประมวลผลภาพทางการแพทย์บนระบบกริดของระบบที่พัฒนาขึ้น	36
4.4	ระยะเวลาในการประมวลผลภาพทางการแพทย์บนระบบกริดของแต่ละเครื่อง	38
4.5	ระยะเวลาในการโอนย้ายข้อมูล	40

สารบัญภาพ

ภาพที่		หน้า
2.1	ระบบจัดเก็บและสื่อสารข้อมูลภาพทางการแพทย์	6
2.2	ภาพทางการแพทย์ (Medical Image) ที่เป็นไปตามมาตรฐาน DICOM	6
2.3	ส่วนประกอบหลักของเมตะดาต้าในภาพทางการแพทย์ตามมาตรฐาน	6
2.4	โครงสร้างพื้นฐานและสิ่งแวดล้อมของระบบกริด	7
2.5	สถาปัตยกรรมของระบบกริด	8
2.6	ส่วนประกอบของโกลบัสทูลคิท	11
2.7	แบบจำลองไปป์ไลน์ (Pipeline) ของการใช้งานไลบรารีของวีทีเค ในการแสดงภาพ	13
2.8	โครงสร้างและสถาปัตยกรรมของโปรแกรม jgViz	15
2.9	สถาปัตยกรรมการประมวลผลภาพแบบไปป์ไลน์บนระบบกริด	16
2.10	กรอบแนวคิดในการวิจัย	18
3.1	รูปแบบการดำเนินการวิจัย	20
3.2	สถาปัตยกรรมของระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูล	24
3.3	สถาปัตยกรรมของระบบการกระจายการประมวลผลภาพทางการแพทย์แบบขนาน	26
3.4	การกระจายการประมวลผลภาพทางการแพทย์แบบขนาน	28
3.5	3D Dental Scans Data source	29
3.6	Visualization Pipeline of 3D Dental Scans	30
3.7	Flowchart ไดอะแกรม ของระบบทั้งหมด	31
4.1	จำนวน CPU ของเครื่องคอมพิวเตอร์ที่ใช้ทดลอง	35
4.2	การใช้งานระบบกริดเพื่อประมวลผล และทรัพยากรทั้งหมดของระบบ	35
ก.1	การใส่ Path VTK	54
ก.2	การใส่ Library VTK	54

สารบัญแผนภูมิ

แผนภูมิที่		หน้า
4.1	แสดงการเปรียบเทียบระยะเวลาการประมวลผลระหว่าง Auto และManual	37
4.2	แสดงการเปรียบเทียบระยะเวลาการประมวลผลระหว่าง Auto และManual ของเครื่องคอมพิวเตอร์แต่ละเครื่อง	39
4.3	แสดงการเปรียบเทียบระยะเวลาการประมวลผลระหว่าง Auto และManual ของการโอนย้ายข้อมูล	40

บทที่ 1
บทนำ

ความเป็นมาและความสำคัญ

ในอดีตกระบวนการวินิจฉัยโรคต้องอาศัยประสบการณ์และความชำนาญของแพทย์ที่ทำการรักษาในการวิเคราะห์และรักษาโรค ซึ่งมีโอกาสผิดพลาดสูง ทำให้มีการพัฒนาเครื่องมืออิเลคทรอนิกส์เข้ามาช่วยในการวินิจฉัยโรคเพื่อความแม่นยำและรวดเร็วมากขึ้น หนึ่งในการพัฒนาเครื่องมือเพื่อช่วยในการวินิจฉัยโรคคือภาพสแกนหรือภาพจากการเอ็กซเรย์ร่างกายของผู้ป่วย ซึ่งทำให้แพทย์สามารถทราบในรายละเอียดต่างๆของร่างกายผู้ป่วย ที่นำไปสู่การค้นคว้าหาแนวทางการรักษาโรคได้ตรงและแม่นยำมากยิ่งขึ้น โดยภาพที่เกิดจากการสแกนหรือภาพเอ็กซเรย์ที่นำมาวินิจฉัยโรคนี้เรียกว่าภาพทางการแพทย์ ซึ่งเป็นชุดภาพที่ประกอบด้วยหลายๆภาพ และมีความละเอียดสูง ในปัจจุบันมีแนวโน้มเพิ่มมากขึ้นที่จะนำภาพทางการแพทย์เข้ามาช่วยวินิจฉัยโรค โดยใช้เครื่องเอ็ม อาร์ ไอ (MRI) เครื่องซีที สแกน (CT Scan) และเครื่องเอ็กซเรย์ (X-Rays) คอมพิวเตอร์ จากผลการสำรวจข้อมูลของศูนย์รัชวิภาเอ็มอาร์ไอ (ยุทธพงษ์ ไชยรส. 2550) พบว่า ในการสแกนร่างกายแต่ละครั้งจะประกอบด้วยภาพดิจิตอลระหว่าง 500 ถึง 1,000 ภาพ และแต่ละภาพมีขนาดระหว่าง 200 กิโลไบท์ (KBytes) ถึง 2 เมกกะไบท์ (MBytes) ซึ่งมีผลทำให้เกิดปัญหาในด้านการจัดเก็บข้อมูล และด้านเวลาของการประมวลผลภาพทางการแพทย์เนื่องจากปริมาณข้อมูลมีขนาดใหญ่และเครื่องที่มีประสิทธิภาพสูงมีราคาแพง ด้วยสาเหตุดังกล่าวนี้ทำให้ได้มีการพัฒนาเทคนิคต่าง ๆ เพื่อเพิ่มประสิทธิภาพของการเข้าถึงข้อมูลขนาดใหญ่และการประมวลผลภาพให้รวดเร็วยิ่งขึ้น เทคนิคที่พัฒนาขึ้นเพื่อเพิ่มประสิทธิภาพในการประมวลผลภาพให้รวดเร็วยิ่งขึ้น มีหลากหลายเทคนิค เช่นการใช้สถาปัตยกรรมของคลัสเตอร์ (Cluster Architecture) (Tang & Qiu. 2009: 238; Zhang, Bajaj, & Blanke. 2001: 51) และเครื่องลูกข่าย-เครื่องแม่ข่าย (Client-Server Architecture) (Beeson, Dwyer, & Thompson. 2005: 7) ที่แบ่งปันทรัพยากรการประมวลผลของแต่ละเครื่องคอมพิวเตอร์ในระบบเครือข่ายให้สามารถประมวลผลภาพทางการแพทย์เพื่อสร้างภาพเสมือนทางการแพทย์ได้อย่างมีประสิทธิภาพ นอกจากเทคนิคที่กล่าวมานี้ ได้มีนักวิจัยได้พัฒนาการประมวลผลภาพทางการแพทย์แบบขนานบนระบบการประมวลผลแบบกริด (Grid Computing) (Aboamama, Muhammad, Kamalrulnizam, and Zainul. 2007; Dutra, Rodrigues, Giraldi, & Schulze. 2007: 381; Li, Gong, & Xing. 2008: 229) ที่สามารถสร้างภาพเสมือนได้อย่างมีประสิทธิภาพ

ในการใช้งานเครื่องคอมพิวเตอร์ส่วนใหญ่จะเป็นการใช้งานขั้นพื้นฐาน ที่ไม่ต้องใช้การประมวลผลที่ซับซ้อน ทำให้หน่วยประมวลผลกลาง (CPU) ถูกใช้งานได้ไม่เต็มประสิทธิภาพ จึงมีกลุ่ม

นักวิจัยและหน่วยงานต่าง ๆ เสนอแนวคิดที่จะนำเอาความสามารถของหน่วยประมวลผลกลางที่ไม่ถูกใช้งานในแต่ละเครื่องคอมพิวเตอร์ที่อยู่ในระบบเครือข่ายมาประมวลผลข้อมูลจำนวนมหาศาลทางด้านวิทยาศาสตร์ โดยแนวคิดดังกล่าวถือว่าเป็นแนวคิดพื้นฐานของการประมวลผลแบบกริด (Grid Computing) ซึ่งได้วิวัฒนาการมาจากการประมวลผลแบบกระจาย (Distributed Systems) โดยวัตถุประสงค์หลักของแนวคิดคือการแบ่งปันทรัพยากรจำนวนมหาศาล (Large Scale Resource Sharing) ที่กระจายและเชื่อมต่อกันผ่านระบบเครือข่ายให้มีประสิทธิภาพในการประมวลผลที่มีความซับซ้อนสูง (High Performance Computing: HPC) จากแนวคิดดังกล่าวทำให้มีกลุ่มนักวิจัยได้อธิบายการประมวลผลแบบกริด (Grid Computing) (Baker, Buyya, & Laforenza. 2002: 1437; Foster, Kesselman, Nick, and Tuecke. 2002; Foster, Kesselman, & Tuecke. 2001: 200; Foster, & Kesselman. 1999: 677) กล่าวคือเป็นเทคโนโลยีทางด้านเครือข่ายและการประมวลผลแบบกระจาย ซึ่งได้ทำการรวบรวมศักยภาพของเครื่องคอมพิวเตอร์ที่แยกกันอยู่ตามที่ต่างๆเข้าด้วยกันด้วยระบบเครือข่ายอินเตอร์เน็ต โดยมีจุดมุ่งหมายเพื่อเพิ่มประสิทธิภาพในการประมวลผล ซึ่งมีผลทำให้ระบบกริดสามารถประมวลผลข้อมูลขนาดใหญ่หรือมีความซับซ้อนสูงได้รวดเร็วยิ่งขึ้น จากแนวคิดที่ท้าทายดังกล่าวทำให้มีการทำวิจัยที่เกี่ยวข้องกับระบบกริดเกิดขึ้นมากมาย โดยมุ่งเน้นประโยชน์ที่ชัดเจนประการหนึ่งของระบบกริดคือด้านของเวลา นั้นคือ คอมพิวเตอร์ 1 เครื่อง อาจใช้เวลาประมวลผลเป็น เดือน หรืออาจเป็น ปีได้ แต่การประมวลผลแบบ กริดนั้นสามารถลดเวลาให้เหลือในระดับอาทิตย์หรือ วัน เท่านั้น (Baker, Buyya, and Laforenza. 2000)

ด้วยสาเหตุของปริมาณข้อมูลภาพดิจิตอลที่ใช้ในการวินิจฉัยโรคมีขนาดและจำนวนมหาศาล ซึ่งก่อให้เกิดปัญหาที่สำคัญคือด้านเวลาในการประมวลผลภาพทางการแพทย์ จากปัญหาที่กล่าวมานี้ทำให้มีความจำเป็นต้องศึกษาและพัฒนาเทคนิคต่าง ๆ เพื่อเข้ามาช่วยเพิ่มประสิทธิภาพในการประมวลผลภาพทางการแพทย์ ซึ่งเทคโนโลยีกริดเป็นหนึ่งในทางเลือกที่จะนำมาใช้เพื่อเพิ่มประสิทธิภาพในการประมวลผลโดยใช้ทรัพยากรของระบบร่วมกัน เช่น ใช้เครื่องคอมพิวเตอร์ในระบบช่วยประมวลผลภาพต่าง ๆ

วัตถุประสงค์ของการวิจัย

โครงการวิจัยนี้มีวัตถุประสงค์ดังนี้

1. ได้โปรแกรมต้นแบบในการประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์ในระบบกริด
2. เพื่อลดระยะเวลาของการประมวลผลภาพทางการแพทย์ที่มีจำนวนมากและขนาดใหญ่ได้อย่างมีประสิทธิภาพ
3. เพื่อให้สามารถใช้ทรัพยากรที่มีอยู่ในระบบได้อย่างมีประสิทธิภาพ

4. เพื่อศึกษาผลของการประมวลผลภาพแบบกระจายในระบบกริดกับระบบเดิม

ขอบเขตการวิจัย

โครงการวิจัยมีขอบเขตประกอบด้วยดังนี้

1. ขอบเขตด้านข้อมูลของการวิจัย: ข้อมูลภาพดิจิตอลทางการแพทย์ (CT Scan)
2. ขอบเขตด้านเนื้อหา
   2.1. โปรแกรม สามารถประมวลผลข้อมูลทางการแพทย์แบบ 3D ได้
   2.2. โปรแกรม สามารถประมวลผลแบบกระจายบนโครงสร้างแบบ Grid ได้
   2.3. ระบบสามารถแบ่งข้อมูลทางการแพทย์แบบเลือกเอง บนระบบ Grid ได้
   2.4. ระบบสามารถแลกเปลี่ยนข้อมูลทางการแพทย์ระหว่างเครื่องลูกข่ายในเครือข่ายได้
   2.5. ระบบสามารถแสดงข้อมูลการใช้งานทรัพยากรของระบบได้ดังนี้
      - การใช้งานซีพียู (CPU)
      - การใช้งานหน่วยความจำ (RAM)
      - การใช้งานเครือข่าย
3. ขอบเขตด้านเวลา: ระยะเวลา 12 เดือน

คำจำกัดความที่ใช้ในงานวิจัย

ระบบกริด (Grid Systems) เป็นรูปแบบของการประมวลผลแบบกระจาย (Distributed Computing) แบบหนึ่งที่ต้องการนำพลังความสามารถในการประมวลผลจากหลาย ๆ องค์กรที่ต่างองค์กรต่างมีการบริหารจัดการทรัพยากร และนโยบายในการรักษาความปลอดภัยที่แตกต่างกันมาใช้งานร่วมกัน การใช้งานทรัพยากรร่วมกันแบบนี้นั้นสอดคล้องกับระบบธุรกิจสมัยใหม่ที่มีการสร้างทีมงานจากบริษัทหลายบริษัทมาทำงานร่วมกันในโครงการเฉพาะกิจบางอย่างเป็นลักษณะของทีมเสมือน (Virtual Team) และเนื่องจากโครงการเฉพาะกิจเหล่านั้นอาจต้องใช้ทรัพยากร ในการประมวลผลและแทนที่จะซื้อหาทรัพยากรมาใหม่ทั้งหมด Grid Computing ก็มีเครื่องมือให้องค์กรสามารถนำเอาทรัพยากรที่แต่ละองค์กรมีอยู่มาใช้งานร่วมกันอย่างปลอดภัยเรียกว่า องค์กรเสมือน (Virtual Organization) เพื่อตอบสนองความต้องการของทีมเสมือน

ซีที แสกน (Computed Tomography: CT Scan) เป็นการตรวจทางการแพทย์ด้วยคลื่นเอกซเรย์ สามารถสร้างภาพตามแนวตัดและแนวขวาง 3 มิติของอวัยวะที่ต้องการตรวจวินิจฉัยทางการแพทย์

โดยใช้คอมพิวเตอร์ความละเอียดสูงในการแปลงสัญญาณภาพ คุณภาพของภาพจะชัดเจนกว่าการตรวจอัลตร้าซาวด์ แต่ด้อยกว่าการสร้างภาพด้วยเรโซแนนซ์แม่เหล็กหรือ เอ็ม อาร์ ไอ (MRI)

ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่จะได้รับจากงานวิจัยชิ้นนี้ประกอบด้วย ด้านวิชาการ ด้านเศรษฐกิจ/พาณิชย์ ด้านสังคมและชุมชน รวมถึงการเผยแพร่ในวารสาร และหน่วยงานที่นำผลการวิจัยไปใช้ประโยชน์

ตารางที่ 1.1 แสดงประโยชน์ที่ได้รับจากงานวิจัย

ประโยชน์ที่คาดว่าจะได้รับ	กลุ่มผู้ใช้ที่เกี่ยวข้อง
1. ระบบซอฟต์แวร์แอพพลิเคชั่นการประมวลผลภาพทางการแพทย์ที่มีความละเอียดสูง 2. ลดการนำเข้าระบบซอฟต์แวร์แอพพลิเคชั่นจากต่างประเทศ 3. การแลกเปลี่ยนประสบการณ์และเรียนรู้ระหว่างนายแพทย์ นักศึกษาแพทย์และผู้เชี่ยวชาญทางการแพทย์ 4. นำเสนอผลงานวิจัยในการประชุมวิชา การระดับชาติหรือนานาชาติ 1 ผลงาน	1. ประชาชนชาวไทยทั่วไป 2. หน่วยงานทางการแพทย์ 3. นายแพทย์ และผู้เชี่ยวชาญทางการแพทย์

บทที่ 2
แนวคิด ทฤษฎี เอกสารและงานวิจัยที่เกี่ยวข้อง

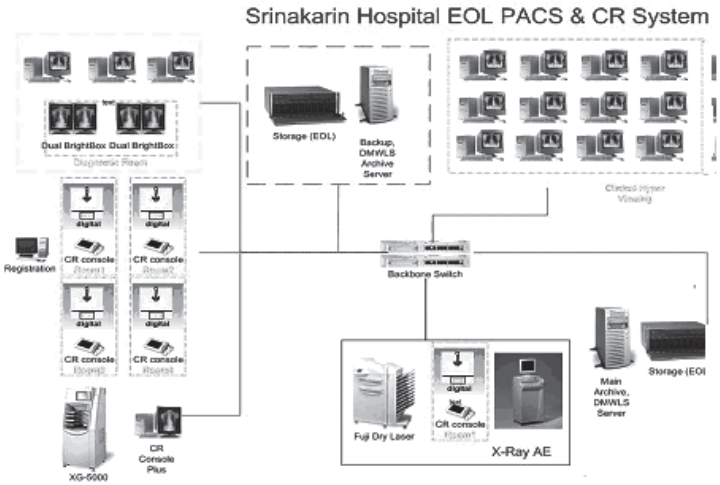
ปัจจุบันมีการใช้เครื่องมืออิเล็กทรอนิกส์โดยเฉพาะระบบภาพดิจิตอลเข้ามาช่วยในกระบวนการวินิจฉัยหาสาเหตุของโรคอย่างแพร่หลาย ซึ่งทำให้การวินิจฉัยโรคเป็นไปอย่างมีประสิทธิภาพและถูกต้องมากยิ่งขึ้น อย่างไรก็ตามในกระบวนการวินิจฉัยโรคโดยใช้ภาพทางการแพทย์ (Medical Images) ซึ่งเป็นภาพที่ได้จากการผ่านกระบวนการรังสีวิทยาเพื่อใช้ในการตรวจสอบร่างกายของคนไข้ เป็นสิ่งที่มีความสำคัญมากทั้งนี้เนื่องจากภาพทางการแพทย์ที่ได้มีความละเอียดสูงและสามารถแสดงรายละเอียดของร่างกายคนไข้อย่างชัดเจน ในกระบวนการได้มาของภาพทางการแพทย์มีความจำเป็นต้องใช้เครื่องคอมพิวเตอร์ที่มีประสิทธิภาพสูงในการประมวลผลภาพทางการแพทย์ เพราะภาพดิจิตอลที่ได้จากเครื่องมืออิเล็กทรอนิกส์มีปริมาณมากและขนาดใหญ่ ดังนั้นได้มีกลุ่มนักวิจัยต่าง ๆ ได้คิดค้นเทคนิคใหม่ ๆ และหลากหลาย ที่ช่วยในการประมวลผลภาพทางการแพทย์ให้รวดเร็วยิ่งขึ้น และใช้ต้นทุนที่ถูกลง

มาตรฐานของการจัดการภาพทางแพทย์

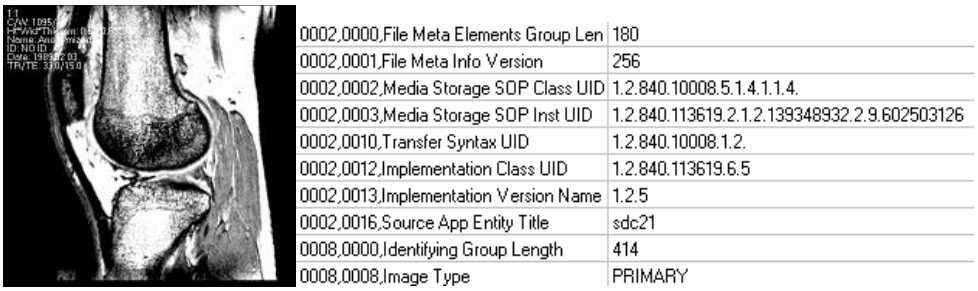
ในการตรวจสอบร่างกายคนไข้เพื่อวินิจฉัยโรคในปัจจุบันมีการใช้เครื่องมืออิเล็กทรอนิกส์ที่หลากหลาย เช่น เครื่อง ซีที แสกน (CT Scan) เอ็มอาร์ไอ แสกน (MRI Scan) และอัลตราซาวด์ (Ultrasound) เข้ามาแสกนร่างกายคนไข้เพื่อนำภาพทางรังสีวิทยาไปวิเคราะห์โดยแพทย์ผู้เชี่ยวชาญ ซึ่งในการบริหารจัดการรูปภาพทางการแพทย์ได้แก่ระบบจัดเก็บและสื่อสารข้อมูลภาพทางการแพทย์ (Picture Archiving and Communication System) หรือพีเอซีเอส (PACS) (ระบบ PACS. 2017; Picture Archiving and Communication System: PACs. 2005; เพชรากร หาญพาณิชย์. 2007: 115) โดยระบบดังกล่าวประกอบด้วยเครื่องกำเนิดภาพ เช่น เครื่องถ่ายภาพกำทอนแม่เหล็ก เครื่องเอกซเรย์คอมพิวเตอร์ เครื่องตรวจคลื่นเสียงความถี่สูง และอุปกรณ์เก็บภาพทางการแพทย์ ภาพที่ได้จากเครื่องกำเนิดภาพจะถูกนำไปเก็บยังอุปกรณ์จัดเก็บ เมื่อแพทย์ผู้เชี่ยวชาญต้องการภาพเพื่อวินิจฉัยโรค ภาพเหล่านี้จะถูกส่งผ่านระบบเครือข่ายภายใต้มาตรฐาน DICOM (Digital Imaging and Communications in Medicine) (NEMA. Digital and Communication in Medicine (DICOM). 2008) โดยภาพที่ 2.1 เป็นตัวอย่างของระบบพีเอซีเอส

สำหรับภาพทางการแพทย์ (Medical Image) ที่เป็นไปตามมาตรฐาน DICOM ประกอบด้วย 2 ส่วน (ยุทธพงษ์ ไชยรส. 2550) คือ 1) ข้อมูลภาพและ 2) ข้อมูลเมตะดาต้า (Meta Data) ซึ่งข้อมูล

เมตะดาต้า ดังภาพที่ 2.2 แสดงการบรรยายถึงข้อมูลคนไข้ รายละเอียดของภาพและรายละเอียดของฮาร์ดแวร์ ซึ่งเป็นที่มาของภาพ 2.3 ที่แสดงองค์ประกอบของข้อมูลเมตะดาต้า

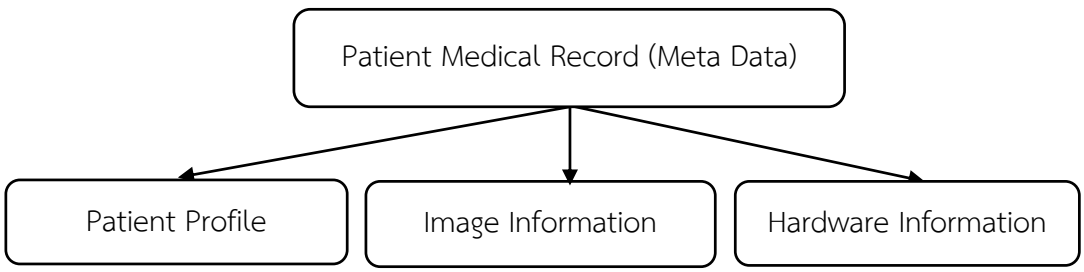


ภาพที่ 2.1 ระบบจัดเก็บและสื่อสารข้อมูลภาพทางการแพทย์ (เพชรากร หาญพาณิชย์. 2007: 115)



0002,0000,File Meta Elements Group Len	180
0002,0001,File Meta Info Version	256
0002,0002,Media Storage SOP Class UID	1.2.840.10008.5.1.4.1.1.4.
0002,0003,Media Storage SOP Inst UID	1.2.840.113619.2.1.2.139348932.2.9.602503126
0002,0010,Transfer Syntax UID	1.2.840.10008.1.2.
0002,0012,Implementation Class UID	1.2.840.113619.6.5
0002,0013,Implementation Version Name	1.2.5
0002,0016,Source App Entity Title	sdc21
0008,0000,Identifying Group Length	414
0008,0008,Image Type	PRIMARY

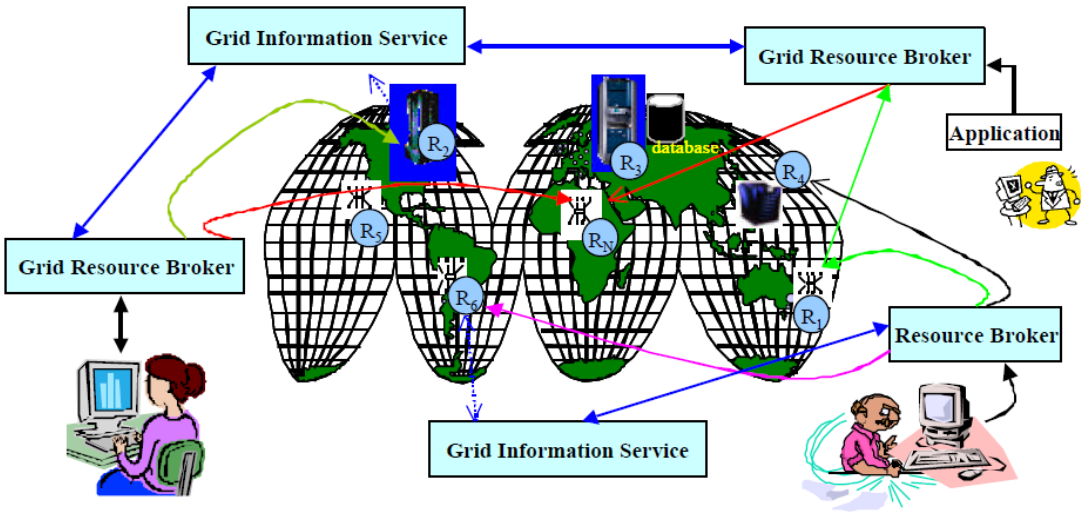
ภาพที่ 2.2 ภาพทางการแพทย์ (Medical Image) ที่เป็นไปตามมาตรฐาน DICOM (ยุทธพงษ์ ไชยรส. 2550)



ภาพที่ 2.3 ส่วนประกอบหลักของเมตะดาต้าในภาพทางการแพทย์ตามมาตรฐาน DICOM (ยุทธพงษ์ ไชยรส. 2550)

เทคโนโลยีกริด (Grid Technology)

เทคโนโลยีกริด (Foster, & Kesselman. 1999: 677; Foster, Kesselman, & Tuecke. 2001: 200) เป็นระบบประมวลผลแบบกระจายขนาดใหญ่ที่ได้รวบรวมเอาทรัพยากรจากองค์กรต่างๆเข้าด้วยกัน โดยการเชื่อมต่อกับเครือข่ายสาธารณะ (Internet) ซึ่งเป็นการใช้ทรัพยากรให้เกิดประโยชน์สูงสุด จากแนวคิดดังกล่าวเป็นการสร้างองค์กรเสมือน (Virtual Organization) ที่ได้รวบรวมทรัพยากรคอมพิวเตอร์ที่ได้กระจายตามสภาพภูมิศาสตร์ต่าง ๆ (Geographically Distributed) เข้าด้วยกัน โดยทรัพยากรคอมพิวเตอร์ เช่น หน่วยประมวลผล (Processing Unit) หน่วยเก็บข้อมูล (Data Storage) ระบบเครือข่าย (Computer Network) และโปรแกรมประยุกต์ต่าง ๆ (Application) ที่กระจายตามองค์กรต่าง ๆ เมื่อนำทรัพยากรเหล่านี้เชื่อมต่อเข้ามาในระบบกริดแล้ว ทรัพยากรทั้งหมดที่เชื่อมต่อเข้ามาจะเป็นส่วนหนึ่งของระบบกริดและสามารถถูกนำไปใช้โดยองค์กรอื่นได้ สำหรับการบริหารจัดการระบบกริดต้องสามารถบริหารจัดการทรัพยากรที่หลากหลายกันผ่านระบบเครือข่ายอินเตอร์เน็ต ดังภาพที่ 2.4 จากคุณสมบัติดังกล่าวทำให้มีการนำโครงสร้างแบบกริดไปประมวลผลโปรแกรมเพื่อแก้ปัญหาที่มีความซับซ้อนมากๆ ตลอดจนมีขนาดของข้อมูลขนาดใหญ่



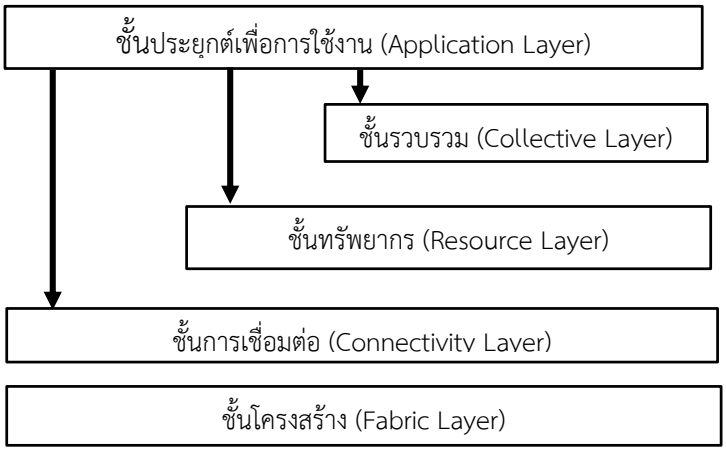
ภาพที่ 2.4 โครงสร้างพื้นฐานและสิ่งแวดล้อมของระบบกริด (Buyya & Murshed. 2002)

จากคุณสมบัติของเทคโนโลยีกริดทำให้มีการทำวิจัยที่หลากหลายมิติ ซึ่งหนึ่งในประเด็นที่น่าสนใจคือการจัดสรรการใช้งานทรัพยากรในระบบให้ได้อย่างมีประสิทธิภาพ โดยเฉพาะในส่วนของหน่วยประมวลผล เพื่อลดระยะเวลาของการประมวลผลข้อมูลที่มีขนาดใหญ่และซับซ้อนสูง ทำให้มีกลุ่มนักวิจัย เช่น (Dalibor & Hana. 2008; Lorpunmanee, Mohd Noor, Abdul Hanan, &

Chompoo-inwai. 2007: 314; Rafsanjani & Bardsiri. 2012: 37; Kamalam & Bhaskaran. 2010: 24) ได้พัฒนาระบบการจัดสรรการใช้งานทรัพยากร โดยเฉพาะหน่วยประมวลผลกลางในระบบกริดให้มีประสิทธิภาพ จากผลของการวิจัยดังกล่าวทำให้มีกลุ่มนักวิจัยสนใจในการนำข้อมูลภาพทางการแพทย์ซึ่งเป็นข้อมูลที่มีขนาดใหญ่และซับซ้อนสูง ตลอดจนมีความสำคัญเนื่องจากต้องใช้เป็นเครื่องมือในการวินิจฉัยโรคของแพทย์ผู้เชี่ยวชาญ มาทำการวิจัยเพื่อเพิ่มประสิทธิภาพในการประมวลผลบนระบบกริด เช่น (Aboamama, Muhammad, Kamalrulnizam, & Zainul. 2007:567; Dutra, Rodrigues, Giraldi, & Schulze. 2007:381)

สถาปัตยกรรมของระบบกริด

แนวคิดเรื่องกริดเกิดขึ้นกลางทศวรรษที่ 1990 เพื่อแก้ปัญหาที่มีความซับซ้อนสูง โดยอาศัยการแบ่งปันทรัพยากรในระบบเครือข่ายที่อยู่ในที่ต่าง ๆ เช่น หน่วยประมวลผล (CPU) ของเครื่องคอมพิวเตอร์ให้มาช่วยประมวลผล แรงจูงใจในการสร้างระบบกริดคือการพัฒนาระบบประมวลผลประสิทธิภาพสูง (High Performance Computing) ที่สามารถแบ่งปันทรัพยากรต่างๆ ไม่ว่าจะเป็นหน่วยประมวลผลกลาง (CPU) ของคอมพิวเตอร์ การจัดเก็บข้อมูล (Storage) เครือข่าย (Network) และอุปกรณ์คำนวณขนาดเล็ก (Small Devices) ที่กระจายอยู่ตามแหล่งต่างๆ ซึ่งไม่มีข้อจำกัดในเรื่องของภูมิศาสตร์ โดยอาศัยการเชื่อมต่อผ่านระบบอินเตอร์เน็ต (Internet) จากแนวคิดดังกล่าวทำให้มีการนำเสนอโครงสร้างแบบองค์กรเสมือน (Virtual Organizations) (Foster, Kesselman, & Tuecke. 2001: 200) เพื่ออธิบายหลักการทำงานของระบบกริด ซึ่งสถาปัตยกรรมของระบบกริดได้แบ่งรูปแบบออกเป็นลำดับชั้น และแต่ละลำดับชั้นมีการทำงานประสานกัน เพื่อให้ง่ายแก่ความเข้าใจ สามารถอธิบายได้ดังภาพที่ 2.5



ภาพที่ 2.5 สถาปัตยกรรมของระบบกริด (Foster, Kesselman, & Tuecke. 2001: 200; Foster, Zhao, Raicu, & Lu. 2008:1)

จากภาพที่ 2.5 องค์ประกอบในแต่ละลำดับชั้นจะมีคุณลักษณะพื้นฐานร่วมกัน และในแต่ละลำดับชั้นก็จะสร้างรูปแบบเป็นของตนเองโดยอาศัยข้อมูลของลำดับชั้นที่ต่ำกว่าได้ โดยรายละเอียดของแต่ละลำดับชั้นสามารถอธิบายได้ดังนี้ (Foster, Kesselman, & Tuecke. 2001: 200; Foster, Zhao, Raicu, & Lu. 2008:1)

1. ชั้นโครงสร้าง (Fabric Layer) เป็นชั้นที่รวบรวมทรัพยากรที่ต้องการใช้ร่วมกัน เช่น หน่วยประมวลกลางของเครื่องคอมพิวเตอร์ การจัดเก็บข้อมูล (Storage) ระบบเครือข่าย (Network) อุปกรณ์ต่อพ่วงต่างๆ (Small Devices) และระบบคอมพิวเตอร์คลัสเตอร์ (Cluster System) ตลอดจนระบบบริหารจัดการทรัพยากรคอมพิวเตอร์แบบกระจาย เช่น พีบีเอส (PBS), คอนดอล์ (Condor) เป็นต้น

2. ชั้นการเชื่อมต่อ (Connectivity Layer) เป็นชั้นที่เกี่ยวข้องกับการติดต่อสื่อสารและการยืนยันตัวตนของผู้ใช้งาน ซึ่งเป็นกระบวนการที่สำคัญสำหรับการติดต่อสื่อสารให้มั่นคงปลอดภัยของระบบกริด ในการใช้งานทรัพยากรของระบบผู้ใช้ต้องยืนยันตัวตน เพื่อให้ระบบสามารถพิสูจน์ทราบสิทธิ์การใช้งานของผู้ใช้และทรัพยากรของระบบได้ โดยระบบกริดได้อาศัยพื้นฐานของมาตรฐานที่มีอยู่ในปัจจุบันทั้งทางด้านการติดต่อสื่อสารและการใช้งานทรัพยากรของระบบให้มั่นคงปลอดภัย ในกำหนดการเข้าไปใช้งานระบบของผู้ใช้ ทำให้มีการพัฒนาการติดต่อสื่อสารที่มีการรักษาความมั่นคงปลอดภัยบนระบบกริดผ่านเครือข่ายอินเทอร์เน็ต เพื่อให้สามารถใช้ทรัพยากรจากแหล่งต่าง ๆ ได้ โดยไม่กระทบกับโครงสร้างของระบบความมั่นคงปลอดภัยในองค์กรที่ทรัพยากรติดตั้งอยู่

3. ชั้นทรัพยากร (Resource Layer) ถูกสร้างให้อยู่บนชั้นการเชื่อมต่อ ในการทำงานของชั้นนี้ได้ถูกกำหนดโดยโพรโทคอลการติดต่อสื่อสารและการยืนยันตัวตนของผู้ใช้ ทั้งนี้เพื่อให้ผู้ใช้สามารถใช้ทรัพยากร เช่น คอมพิวเตอร์ส่วนบุคคล หรือแม้แต่คอมพิวเตอร์ในระบบคอมพิวเตอร์คลัสเตอร์ได้อย่างปลอดภัย นอกจากนี้โพรโทคอลที่สร้างขึ้นมายังช่วยในการกำหนดการเริ่มต้นการทำงานของระบบ การตรวจสอบสถานะต่าง ๆ การควบคุมดูแล ตลอดจนจัดทำระบบบัญชีรายชื่อเพื่อแสดงสถานะการใช้งานของผู้ใช้งานในทรัพยากรแต่ละส่วน ซึ่งในการเข้าใช้งานทรัพยากรในแต่ละส่วนของระบบ ชั้นทรัพยากรจะใช้โพรโทคอลทำการร้องขอการใช้ทรัพยากรผ่านชั้นโครงสร้างเพื่อเข้าใช้และควบคุมทั้งในส่วนของทรัพยากรส่วนบุคคลและทรัพยากรที่กระจายอยู่ในชั้นโครงสร้าง โดยโพรโทคอลในชั้นนี้สามารถแบ่งออกเป็น 2 ส่วน คือ

3.1 โพรโทคอลข้อมูล (Information Protocols) ถูกสร้างขึ้นเพื่อเก็บรวบรวมข้อมูลที่เกี่ยวกับโครงสร้างและสถานะของทรัพยากร เช่น การกำหนดค่าการติดตั้งของทรัพยากรระบบ ภาระงานในปัจจุบันและนโยบายการใช้งาน

3.2 โพรโทคอลการจัดการ (Management Protocols) ถูกสร้างขึ้นมาเพื่อใช้ในการต่อรองการเข้าใช้งานทรัพยากรร่วมกันของระบบ เช่น ข้อกำหนดของการใช้ทรัพยากร (รวมถึงการจองทรัพยากรในการใช้งานและการประกันคุณภาพของการให้บริการของระบบ) และการดำเนินงานของระบบ

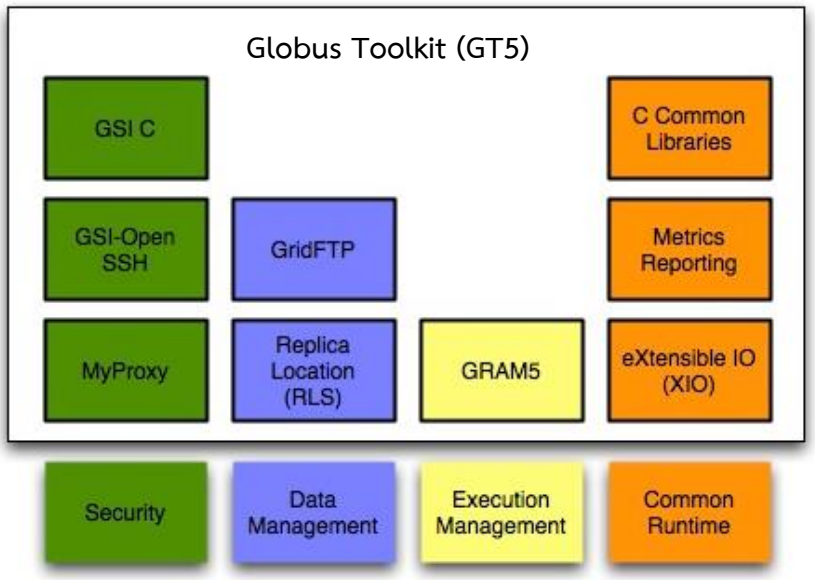
4. ชั้นรวบรวม (Collective Layer) เป็นชั้นที่มีหน้าที่ติดต่อกับทรัพยากรเหมือนกับชั้นทรัพยากร แต่ชั้นรวบรวมนี้ประกอบด้วยโพรโทคอลและการบริการต่าง ๆ รวมถึงหน่วยย่อยอย่างเอพีไอ (API) และเอสดีเค (SDKs) โดยบริการเหล่านี้จะทำหน้าที่รวบรวมทรัพยากรที่หลากหลายรูปแบบ ระบบปฏิบัติการและกระจายอยู่ตามแหล่งต่างๆ ซึ่งในชั้นนี้ได้มีกลุ่มนักวิจัยได้การพัฒนาที่หลากหลายรูปแบบโดยมีวัตถุประสงค์ที่แตกต่างกันไป เช่น กลุ่มนักวิจัยที่ศึกษา AppLeS (Berman, Wolski, Figueira, Schopf, & Shao. 1996; Foster, & Kesselman. 1999: 677) และ Condor-G (Frey, Tannenbaum, Livny, Foster, & Tuecke. 2002: 237) ได้ศึกษาในเรื่องของการจัดสรรงานไปประมวลผลในทรัพยากรที่มีการใช้งานร่วมกันของระบบ การจัดตารางงานของระบบ เป็นต้น

โกลบัสทูลคิท (Globus Toolkit)

ในการพัฒนาระบบกริดนั้นโครงสร้างและทรัพยากรทั้งหมดได้ถูกรวบรวมเข้าด้วยกันทั้งนี้เพื่อประโยชน์ในการแบ่งปันทรัพยากรเหล่านั้นในการประมวลผล ซึ่งซอฟต์แวร์ที่ทำหน้าที่ในการสร้างโครงสร้างเหล่านี้เรียกว่า กริดมิดเดิลแวร์ (Grid Middleware) ปัจจุบันมิดเดิลแวร์ที่มีมาตรฐานและเป็นที่นิยมใช้มากที่สุดได้แก่ โกลบัสทูลคิท (Globus Toolkit) โดยพัฒนาขึ้นในปี ค.ศ. 1998 ในเวอร์ชั่นที่ 1.0 และพัฒนาต่อเนื่องถึงปัจจุบันในเวอร์ชั่นล่าสุด 6.0 ซึ่งโกลบัสทูลคิท (Globus Toolkit: GT) (Welcome to the Globus Toolkit Homepage. 2017) เป็นซอฟต์แวร์มิดเดิลแวร์ที่รวบรวมชุดเครื่องมือที่เป็นพื้นฐานในการสร้างระบบกริด เช่น จีอาร์เอเอ็ม (Globus Resource Allocation Manager: GRAM) ทำหน้าที่ในการจัดสรรทรัพยากร และจีเอสไอ (Grid Security Infrastructure: GSI) ทำหน้าที่ในการตรวจสอบความถูกต้องและยืนยันตัวตนของผู้ใช้งาน (Authentication) ตลอดจนการเข้าใช้ข้อมูลของผู้ใช้ (Access Control) เป็นต้น นอกจากนี้โกลบัสทูลคิทเป็นซอฟต์แวร์มาตราฐานเปิด (Open Source) ที่นักวิจัยสามารถพัฒนาต่อยอดได้อย่างถูกต้องตามกฎหมาย

โกลบัสทูลคิทเวอร์ชั่น 6 (GT6) เป็นเวอร์ชั่นล่าสุดที่ได้พัฒนาขึ้นมาเพื่อรองรับระบบที่มีโครงสร้างแบบกระจาย และได้พัฒนาให้มีความเสถียรมากยิ่งขึ้น โดยโกลบัสทูลคิทได้พัฒนาบนพื้นฐานของสถาปัตยกรรมโอจีเอสเอ (Open Grid Services Architecture: OGSA) (Open Grid Services Architecture. 2017) ซึ่งโครงสร้างของระบบจะประกอบด้วยบริการต่างๆ (Services) ที่สามารถแลกเปลี่ยนข้อมูลระหว่างกันได้และให้บริการทรัพยากรแก่ผู้ใช้เรียกว่า (Service-Oriented Architecture: SOA) โปรแกรมที่พัฒนาขึ้นมาเพื่อใช้งานบนระบบกริดต้องพัฒนาให้อยู่ในรูปแบบของกริดเซอร์วิส (Grid Services)

บนมาตรฐานของ Web Services Resource Framework (WSRF) โดยองค์ประกอบหลักของโกลบัสทูลคิท ประกอบส่วนต่าง ๆ ดังภาพที่ 2.6 ดังนี้ (About the Globus Toolkit. 2017)



ภาพที่ 2.6 ส่วนประกอบของโกลบัสทูลคิท (About the Globus Toolkit. 2017)

1. ด้านการจัดการความมั่นคงของระบบกริด (Security) เกี่ยวกับระบบความมั่นคงปลอดภัยของการใช้งานระบบกริด ประกอบด้วยบริการต่าง ๆ ได้แก่ การบริหารจัดการใบอนุญาตรับรอง (Credential Management) การยืนยันตัวตนและการอนุญาตเข้าไปใช้งานระบบกริด (Authentication, Authorization) การสร้างตัวแทนในการติดต่อ (Delegation) และการให้สิทธิ์เข้าใช้ในทรัพยากรกริด (Community Authorization) โดยโกลบัสทูลคิทได้เตรียมเครื่องมือสำหรับบริการต่าง ๆ เหล่านี้ ประกอบด้วยองค์ประกอบหลัก (Core Component) ที่สำคัญคือ กริด ซีเคียวริที อินฟระสทรัคเชอะ ฟอร์ ซี (Grid Security Infrastructure in C: GSI C) จีเอสไอ-โอเพ่นเอสเอสเอช (GSI-OpenSSH) และมายพร็อกซี่ (MyProxy)

2. ด้านการจัดการข้อมูลของระบบกริด (Data Management) เกี่ยวกับการจัดการข้อมูลต่างๆ ซึ่งประกอบด้วยบริการต่างๆ ได้แก่ กริดเอฟทีพี (GridFTP) ที่ทำหน้าที่ในการส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์ในระบบเครือข่ายของกริด การคัดลอกสำเนาไฟล์ในระบบกริด (Replica Location: RLS)

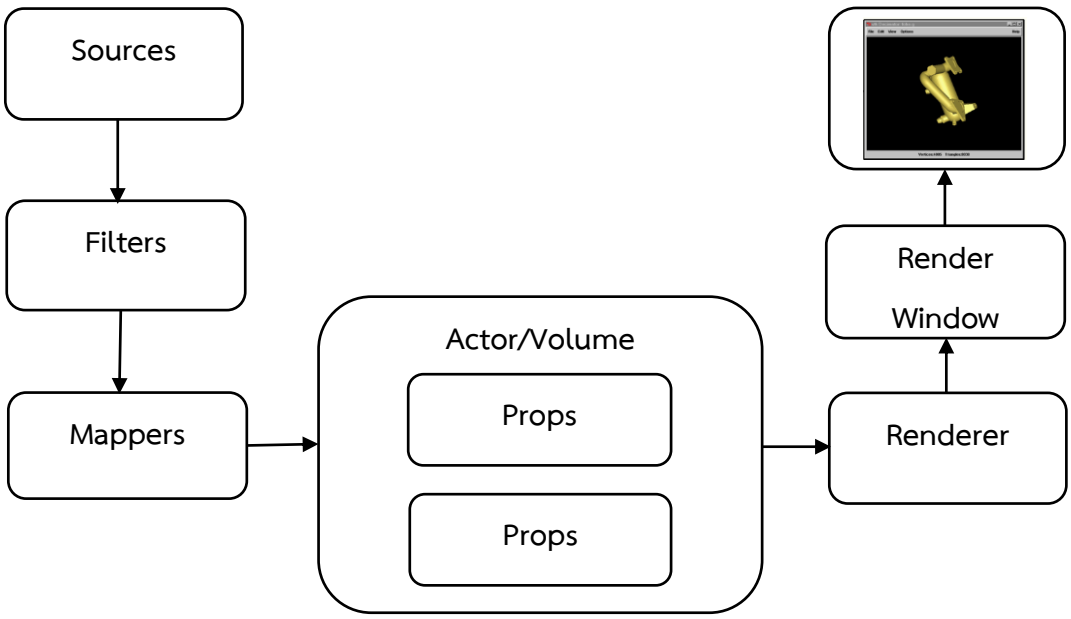
3. ด้านการบริหารจัดการของระบบกริด (Execution Management) เกี่ยวกับการบริหารจัดการทรัพยากรทั้งหมดของระบบกริด การจองหรือมอบหมายทรัพยากรเพื่อดำเนินการประมวลผลตามคำสั่งของผู้ใช้ หรือแม้แต่ยกเลิกทรัพยากรในการประมวลผลจากผู้ใช้ ตลอดจนการตรวจสอบ

สถานะของทรัพยากรในระบบกริด ซึ่งบริการที่สำคัญที่ทำหน้าที่ดังกล่าวคือ จีอาร์เอเอ็ม (The Grid Resource Allocation and Management: GRAM)

4. ด้านการเชื่อมต่อใช้งานทั่วไปของระบบกริด (Common Runtime) โดยโกลบัสทูลคิทได้เตรียมเครื่องมือสำหรับการเชื่อมต่อต่าง ๆ เพื่อให้สามารถพัฒนาโปรแกรมเชื่อมต่อเข้ากับระบบกริดอย่างมีประสิทธิภาพ ซึ่งประกอบด้วย เอ็กซ์ไอโอ (Extensible IO: XIO) ไลบรารี่พื้นฐานของภาษาซี (C Common Libraries) และไลบรารี่ต่าง ๆ เช่น ภาษาจาวา (Java) ภาษาซี (C) และภาษาไพธอน (Python) สำหรับพัฒนากริดเซอร์วิส

วีทีเคทูลคิท (VTK: Visualization Toolkit)

วีทีเคทูลคิท (VTK: Visualization Toolkit) (The Visualization Toolkit (VTK). 2017) เป็นซอฟต์แวร์โอเพนซอร์สที่ได้เตรียมไลบรารีไว้สำหรับงานคอมพิวเตอร์กราฟฟิกส์แบบ 3 มิติ (3D Computer Graphics) การสร้างโมเดล (Modeling) การประมวลผลภาพ (Image Processing) และการสร้างภาพเสมือน (Visualization) จากคุณสมบัติดังกล่าวทำให้มีการนำไลบรารีของวีทีเค (VTK Library) มาพัฒนาซอฟต์แวร์ที่เกี่ยวกับภาพทางการแพทย์ ซึ่งทำให้การพัฒนาซอฟตแวร์เป็นไปอย่างสะดวกและรวดเร็ว ตลอดจนสามารถลดต้นทุนในการพัฒนาได้ ในปัจจุบันได้มีการพัฒนาไลบรารีของวีทีเคให้มีความสามารถในการรองรับภาษาที่หลากหลาย เช่น ภาษาจาวา (Java), ซีพลัสพลัส (C++) ทีซีเอลและทีเค (Tcl/Tk) และไพธอน (Python) ตลอดจนเตรียมฟังก์ชั่นและอัลกอริธึมเกี่ยวกับงานแสดงภาพและประมวลผลภาพให้เลือกใช้มากมาย ในการพัฒนาซอฟต์แวร์ภาพ 3 มิติโดยใช้งานไลบรารีของวีทีเคสามารถพัฒนาในรูปแบบของไปป์ไลน์ (Pipeline) ซึ่งประกอบด้วยแต่ละส่วนและทำงานผ่านเป็นลำดับขั้นโดยจะข้ามขั้นตอนไม่ได้ ดังภาพที่ 2.7



ภาพที่ 2.7 แบบจำลองไปป์ไลน์ (Pipeline) ของการใช้งานไลบรารีของวีทีเค ในการแสดงภาพ

ในการพัฒนาซอฟต์แวร์เพื่อสร้างภาพสามมิติโดยใช้ไลบรารีของวีทีเคนั้น ประกอบด้วยกระบวนการที่สำคัญดังภาพที่ 2.7 กล่าวคือ

ขั้นตอนที่ 1 การนำเข้าข้อมูลภาพ (Sources) ซึ่งข้อมูลภาพแบ่งออกเป็น 2 ประเภทคือ จีโอะเมทริค เรพพริเซนเทเชิน (Geometric Representation) และ อิมเมจ เรพพริเซนเทเชิน (Image Representation)

ขั้นตอนที่ 2 การคัดกรองข้อมูลภาพ (Filters) ซึ่งข้อมูลภาพแบบอิมเมจ เรพพริเซนเทเชิน (Image Representation) จะถูกแปลงให้อยู่ในรูปแบบของ จีโอะเมทริค เรพพริเซนเทเชิน (Geometric Representation)

ขั้นตอนที่ 3 การเตรียมข้อมูลภาพ (Mappers) เพื่อให้เป็นข้อมูลต้นทางในการสร้างภาพสามมิติ โดยจะส่งต่อไปยังส่วนของขั้นตอนที่ 4

ขั้นตอนที่ 4 กระบวนการสร้างภาพเสมือนสามมิติ (Actor/Volume) ซึ่งข้อมูลที่นำมาสร้างภาพต้องผ่านมาจากขั้นตอนที่ 3 โดยข้อมูลภาพต้องมีคุณสมบัติ เช่น ความละเอียดของภาพ สี ขนาดของหน้าตัดภาพ มาโดยละเอียด

ขั้นตอนที่ 5 การประมวลผลเพื่อแสดงภาพเสมือนสามมิติหรือสองมิติ (Renderer)

ขั้นตอนที่ 6 ภาพเสมือนสามมิติ (Render Window) ที่สามารถแสดงออกทางหน้าจอคอมพิวเตอร์

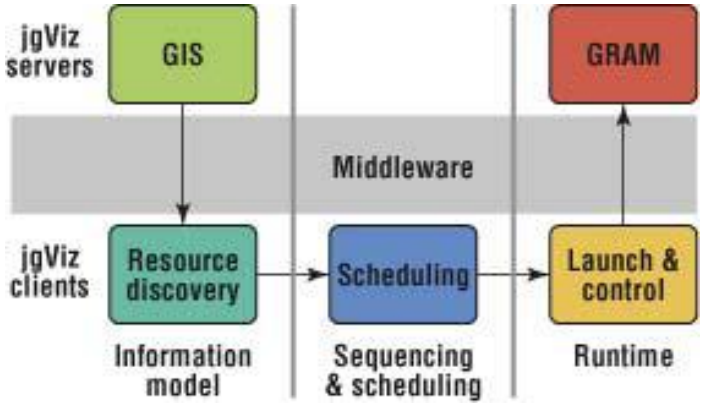
การสร้างภาพวิชวลไลเซชัน (Visualization)

การสร้างภาพวิชวลไลเซชัน (Visualization) เป็นกระบวนการประมวลผลชุดข้อมูลภาพ (Volumetric data) ที่มีความซับซ้อนให้สามารถแสดงผลออกมาเป็นภาพเสมือนสามมิติ โดยการแสดงผลของภาพ สามารถแสดงผลออกเป็น 2 แบบ ได้แก่ การแสดงผลแบบพื้นผิว (Surface Rendering) และการแสดงผลแบบปริมาตร (Volume Rendering) ซึ่งหลักการทั้ง 2 แบบนี้จะทำการแทนที่ชุดข้อมูลด้วยระบบจุดหลายๆ จุดประกอบกันจนเป็นภาพ ในการแสดงภาพเสมือนสามมิติที่มีความต้องการในรายละเอียดของภาพสูง โดยปกติจะใช้การแสดงผลแบบปริมาตร (Volume Rendering) ซึ่งไม่เพียงแต่จะได้ภาพพื้นผิวภาพสามมิติเท่านั้น แต่จะได้ปริมาตรภายในของภาพสามมิติด้วย ข้อมูลที่นำมาประมวลผลแบบปริมาตร จะเป็นข้อมูลทั้งหมดที่เป็นข้อมูลดิบ (Raw Data) ซึ่งจะรวมไปถึงข้อมูลส่วนที่เป็นค่าสี ค่าความเข้ม ความทึบของจุดสี ในแต่ละ โวเซลล์ (Voxel) จากคุณสมบัติในด้านรายละเอียดของภาพ ความถูกต้องครบถ้วนของข้อมูลภาพและความแม่นยำเป็นสำคัญ ดังนั้นทำให้มีการนำเทคนิคของการแสดงผลแบบปริมาตรมาใช้ในอุตสาหกรรมทางการแพทย์ที่มีความต้องการนำภาพสามมิติมาประกอบการวินิจฉัยโรคของแพทย์ผู้เชี่ยวชาญ อย่างไรก็ตามเทคนิคดังกล่าวนี้มีความจำเป็นต้องใช้เครื่องที่มีประสิทธิภาพสูงในการประมวลผลภาพทั้งนี้เนื่องจากต้องนำข้อมูลภาพทั้งหมดที่เป็นข้อมูลดิบ และมีขนาดใหญ่มาประมวลผล

งานวิจัยที่เกี่ยวข้องกับการประมวลผลภาพทางการแพทย์บนระบบกริด

ตลอดระยะเวลาหลายปีที่ผ่านมา การสร้างภาพเสมือนในรูปแบบไปป์ไลน์ (Visualization Pipeline) ได้มีการพัฒนาซอฟต์แวร์เพื่อใช้งานลักษณะดังกล่าวนี้อย่างต่อเนื่อง เช่น OpenDX (Abram & Treinish. 1995: 263) SCIRun (Parker & Johnson. 1995) และ VTK (Schroeder, Martin, and Lorensen. 2004) อย่างไรก็ตามปัญหาที่สำคัญประการหนึ่งของการสร้างภาพเสมือนในรูปแบบไปป์ไลน์คือความต้องการใช้ทรัพยากรของระบบคอมพิวเตอร์อันได้แก่หน่วยความจำ (Memory) และหน่วยประมวลผลกลาง (CPU) เพื่อการประมวลผลภาพที่มีความละเอียดสูงและข้อมูลภาพมีจำนวนมาก (Moreland. 2013: 367) ทำให้มีกลุ่มนักวิจัยได้วิจัยและพัฒนาซอฟต์แวร์การสร้างภาพเสมือนโดยอาศัยการประมวลผลภาพในระบบคอมพิวเตอร์ที่มีโครงสร้างของระบบคอมพิวเตอร์ที่หลากหลาย เช่น ระบบคลัสเตอร์ Rodrigues, Balan, Zaina, & Traina. 2015: 79; Tang & Qiu. 2009: 438) หรือแม้แต่ระบบกริด (Kingsy, Priyavarshini, Rajasri, & Rajeshwari. 2012; Yang, Chen, & Yang. 2010: 1127; Grace, Manimegalai, & Kumar. 2014: 144)

จากคุณสมบัติของระบบกริด (Grid Computing) ที่สามารถแบ่งปันทรัพยากรต่าง ๆ ที่กระจายในองค์กรเสมือนให้กับผู้ใช้เพื่อประมวลผลข้อมูลหรือโปรแกรมได้อย่างมีประสิทธิภาพ ทำให้มีกลุ่มนักวิจัยที่สนใจในการศึกษาวิจัยการประมวลผลภาพทางการแพทย์บนระบบกริด ในปี ค.ศ. 2007 กลุ่มนักวิจัย (Fewings & John. 2007: 1) ทำการศึกษาวิจัยเกี่ยวกับการประมวลผลภาพบนระบบกริด และได้พัฒนาโปรแกรมชื่อ jgViz ซึ่งใช้ภาษาจาวาพัฒนาขึ้นบนโกลบัสทูลคิท (Globus Toolkit) โดยมีวัตถุประสงค์เพื่อให้โปรแกรมสามารถประมวลผลภาพที่มีข้อมูลภาพจากแหล่งข้อมูลในที่ต่าง ๆ บนระบบกริด และสามารถใช้ทรัพยากรของระบบกริดมาช่วยประมวลผลภาพอย่างมีประสิทธิภาพ ในการออกแบบโปรแกรม ระบบโครงสร้าง และสถาปัตยกรรมของระบบ มีรายละเอียดดังภาพที่ 2.8

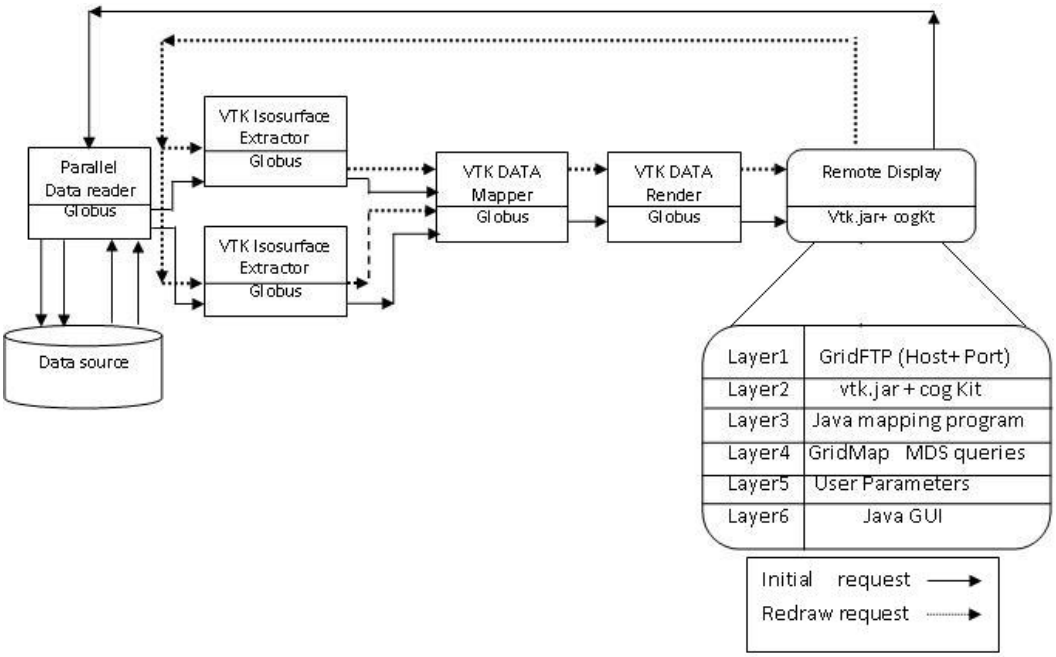


ภาพที่ 2.8 โครงสร้างและสถาปัตยกรรมของโปรแกรม jgViz (Fewings & John. 2007: 1)

จากภาพโปรแกรมระบบที่พัฒนาขึ้นประกอบด้วย 2 ส่วน ได้แก่ ฝั่งผู้ร้องขอ (Client-Side) และผู้ให้บริการ (Server-Side) โดยมีโกลบัสทูลคิทซึ่งเป็นกริดมิดเดิลแวร์ทำหน้าที่ในการสร้างโครงสร้างของระบบกริด ในการพัฒนาโปรแกรม jgViz ของนักวิจัยกลุ่มนี้ได้ใช้คุณสมบัติพื้นฐานของโกลบัสทูลคิท เช่น GRAM, GridFTP, GIS (Grid Information Service) เข้ามาสนุบสนุนการทำงานของโปรแกรม ซึ่งผลของการทดลองสามารถสรุปได้ว่า การประมวลผลภาพโดยทรัพยากรของระบบกริดสามารถทำงานได้อย่างมีประสิทธิภาพ ทั้งในรูปแบบของการใช้งานหน่วยประมวลผลกลาง (CPU) และหน่วยความจำ (Memory) ในปีเดียวกันกลุ่มนักวิจัย (Aboamama, Muhammad, Kamalrulnizam, and Zainul. 2007; Aboamama, Muhammad, Kamalrulnizam, and Zainul. 2007: 114; Aboamama, Muhammad, Kamalrulnizam, and Zainul. 2007: 567) ได้ทำการวิจัยการประมวลผลภาพทางการแพทย์บนระบบกริด (Grid Systems) ซึ่งนักวิจัยกลุ่มนี้ได้มีการพัฒนาโปรแกรมการประมวลผลภาพโดยใช้ซอฟต์แวร์โอเพนซอร์ส The Visualization Toolkit (VTK) เป็นฐานในการพัฒนา และใช้โปรแกรม โกลบัสทูลคิท (Globus Toolkit) เป็นกริดมิดเดิลแวร์ทำหน้าที่สร้างโครงสร้างของระบบกริด ตลอดจนให้บริการที่สำคัญของระบบ กลุ่มนักวิจัยได้นำเสนอเทคนิคการ

ประมวลผลภาพแบบขนานในระบบกริด ดังภาพที่ 2.9 แสดงสถาปัตยกรรมของเทคนิคที่กลุ่มนักวิจัยได้นำเสนอ โดยผลการทดลองแสดงการเปรียบเทียบระหว่างการประมวลผลแบบขนานและแบบปกติ ซึ่งเทคนิคที่กลุ่มนักวิจัยเสนอสามารถประมวลผลภาพทางการแพทย์ได้อย่างมีประสิทธิภาพ

ในประเด็นของประสิทธิภาพการกระจายงานเพื่อการประมวลผลภาพทางการแพทย์ในระบบกริด นอกจากกลุ่มนักวิจัยที่ได้กล่าวมาแล้ว ในปีเดียวกันได้มีกลุ่มนักวิจัย (Dutra, Rodrigues, Giraldi, & Schulze. 2007: 381) ทำการศึกษาวิจัยโดยเน้นในเรื่องของการเพิ่มประสิทธิภาพและความเร็วในการประมวลผล ซึ่งกลุ่มนักวิจัยได้พัฒนาโปรแกรมต้นแบบโดยอาศัย The Visualization Toolkit (VTK) เป็นฐานในการพัฒนา และใช้โปรแกรมโกลบัสทูลคิท (Globus Toolkit) เป็นกริดมิดเดิลแวร์ ซึ่งผลการทดลองเมื่อเพิ่มจำนวนเครื่องคอมพิวเตอร์ (Node) เวลาที่ใช้ในการประมวลผลภาพลดลงอย่างมีนัยยะ การประมวลผลภาพโดยใช้ทรัพยากรในระบบกริดอย่างมีประสิทธิภาพและสามารถกระจายการทำงานไปยังเครื่องคอมพิวเตอร์ต่าง ๆ ในระบบกริดนั้น นักวิจัยบางกลุ่มได้พัฒนาเทคนิคที่ทำให้สามารถค้นหาเครื่องคอมพิวเตอร์ในระบบกริดที่สามารถประมวลผลได้ เช่น (Li, Gong, & Xing. 2008: 229) ได้นำเสนอเทคนิค Web Services เข้ามาช่วยในการค้นหาทรัพยากรเพื่อประมวลผลภาพ ซึ่งในปี 2009 นักวิจัยชื่อ Liu Yi และ Gao Shu (Liu & Gao. 2009: 615) ได้พัฒนา

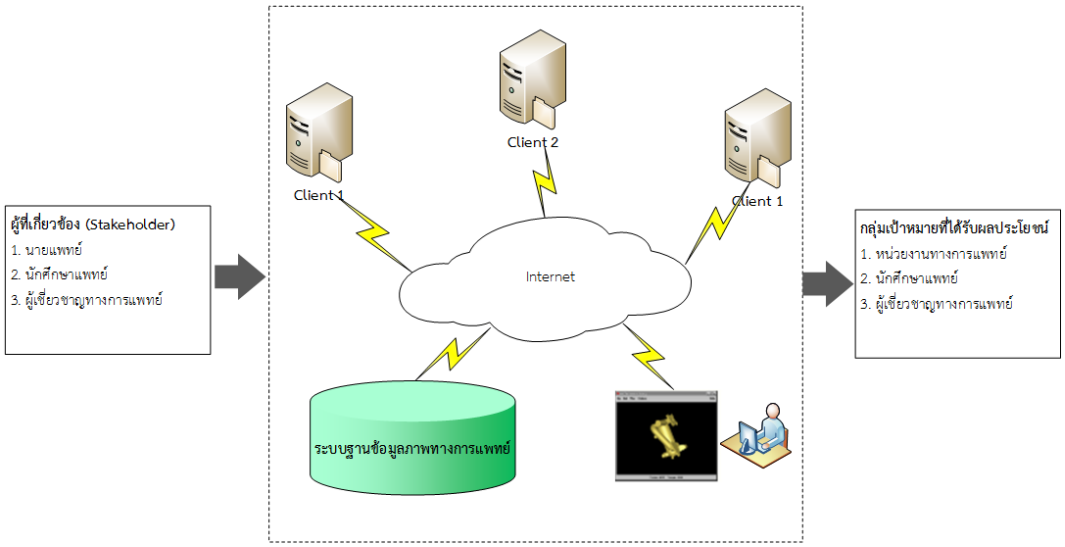


ภาพที่ 2.9 สถาปัตยกรรมการประมวลผลภาพแบบไปป์ไลน์บนระบบกริด (Aboamama, Muhammad, Kamalrulnizam, & Zainul. 2007: 114)

โปรแกรมระบบการประมวลผลภาพโดยนักวิจัยได้นำเสนอแนวคิดของการประมวลผลภาพโดยใช้ Web Services เพื่อประมวลผลบนระบบกริด และได้เรียกชื่อว่า Web Services Resource Framework (WSRF) ระบบดังกล่าวจะทำการเลือกเครื่องคอมพิวเตอร์แบบอัตโนมัติในการประมวลผลภาพโดยอาศัยข้อมูลของการทำงานของเครื่องคอมพิวเตอร์ ซึ่งจุดเด่นของงานวิจัยนี้คือระบบจะทำการเลือกเครื่องคอมพิวเตอร์แบบอัตโนมัติและสามารถประมวลผลภาพทางการแพทย์แบบขนานของข้อมูลที่มีขนาดใหญ่และต้องอาศัยเครื่องคอมพิวเตอร์ที่มีสมรรถนะสูงในการประมวลผลให้รวดเร็ว จากปัญหาประสิทธิภาพของการประมวลผล ความเร็วในการประมวลผล และการโอนย้ายข้อมูลขนาดใหญ่ประกอบกับเครื่องคอมพิวเตอร์สมรรถนะสูงมีราคาที่สูง จึงทำให้ได้มีกลุ่มนักวิจัยได้ทำการพัฒนาโปรแกรมระบบประมวลผลภาพทางการแพทย์ที่หลากหลายแนวคิด

ปี 2010 กลุ่มนักวิจัย (Yang, Chen, & Yang. 2010: 1127) ได้นำเสนอเทคนิคที่ช่วยให้การโอนย้ายข้อมูลภาพในระบบกริดได้รวดเร็วยิ่งขึ้น โดยกลุ่มนักวิจัยได้เสนอ MIFAS (Medical Image File Accessing System) เพื่อช่วยลดระยะเวลาในการโอนย้ายข้อมูลระหว่างเครื่องแม่ข่ายและเครื่องลูกข่ายในระบบกริด ซึ่งเทคนิคต่าง ๆ ในการประมวลผลภาพแบบขนานได้มีการนำเสนอโดยกลุ่มนักวิจัย (Mohammed, Muhammad, Aboamama, Yousif & Manhal. 2013: 223) นอกจากนี้การพัฒนาเครื่องมือที่ช่วยในการบริหารจัดการข้อมูลขนาดใหญ่ที่ชื่อ ฮาดูป (Hadoop) ทำให้มีกลุ่มนักวิจัย (Grace, Manimegalai, & Kumar. 2014: 144) ได้พัฒนาการประมวลผลภาพทางการแพทย์โดยใช้คุณสมบัติเด่นของฮาดูปในเรื่องของการบริหารจัดการข้อมูลที่มีขนาดใหญ่ได้อย่างมีประสิทธิภาพและระบบกริดซึ่งมีคุณสมบัติเด่นในการใช้ทรัพยากรร่วมกัน โดยผลของการทดลองได้แสดงให้เห็นว่าการนำเทคโนโลยีทั้ง 2 มาใช้ร่วมกันทำให้การใช้งานทรัพยากรการประมวลผลได้อย่างมีประสิทธิภาพ และระยะเวลาในการประมวลผลรวดเร็วยิ่งขึ้น อย่างไรก็ตามการจัดสรรทรัพยากรของระบบกริดเพื่อประมวลผลภาพทางการแพทย์ให้มีประสิทธิภาพสูงสุด และประมวลผลให้รวดเร็วยิ่งขึ้น ยังเป็นประเด็นที่สำคัญและน่าสนใจซึ่งมีผลทำให้มีกลุ่มนักวิจัยได้ศึกษาวิจัยเพื่อคิดค้นหาเทคนิคใหม่ต่าง ๆ ตลอดมาจนถึงปัจจุบัน

กรอบแนวคิดในการวิจัย (Conceptual Framework)



ภาพที่ 2.10 กรอบแนวคิดในการวิจัย

ในการพัฒนางานวิจัยประกอบด้วย 3 ส่วนคือ 1. ผู้ที่เกี่ยวข้อง ได้แก่ นายแพทย์ นักศึกษาแพทย์และผู้เชี่ยวชาญทางการแพทย์ ในส่วนนี้จะเป็นผู้ให้ข้อมูลเกี่ยวกับภาพทางการแพทย์ที่สามารถเอาไปวิเคราะห์เพื่อวินิจฉัยโรคได้และกลุ่มคนเหล่านี้จะเป็นผู้ประเมินคุณภาพของงานวิจัย 2. ระบบการประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์บนระบบกริด ทำการวิจัยและพัฒนาโปรแกรมโดยใช้ข้อมูลภาพถ่ายของเครื่อง CT Scan ซึ่งจะได้โปรแกรมต้นแบบที่สามารถกระจายการประมวลผลไปยังเครื่องคอมพิวเตอร์ต่าง ๆ ได้ และเครื่องแม่ข่ายสามารถประมวลผลภาพถ่ายทางการแพทย์ได้อย่างสมบูรณ์ 3. กลุ่มเป้าหมายที่ได้รับผลประโยชน์ ได้แก่หน่วยงานทางการแพทย์ นายแพทย์และผู้เชี่ยวชาญทางการแพทย์ ซึ่งกลุ่มเหล่านี้สามารถนำเอาโปรแกรมต้นแบบไปใช้งานได้

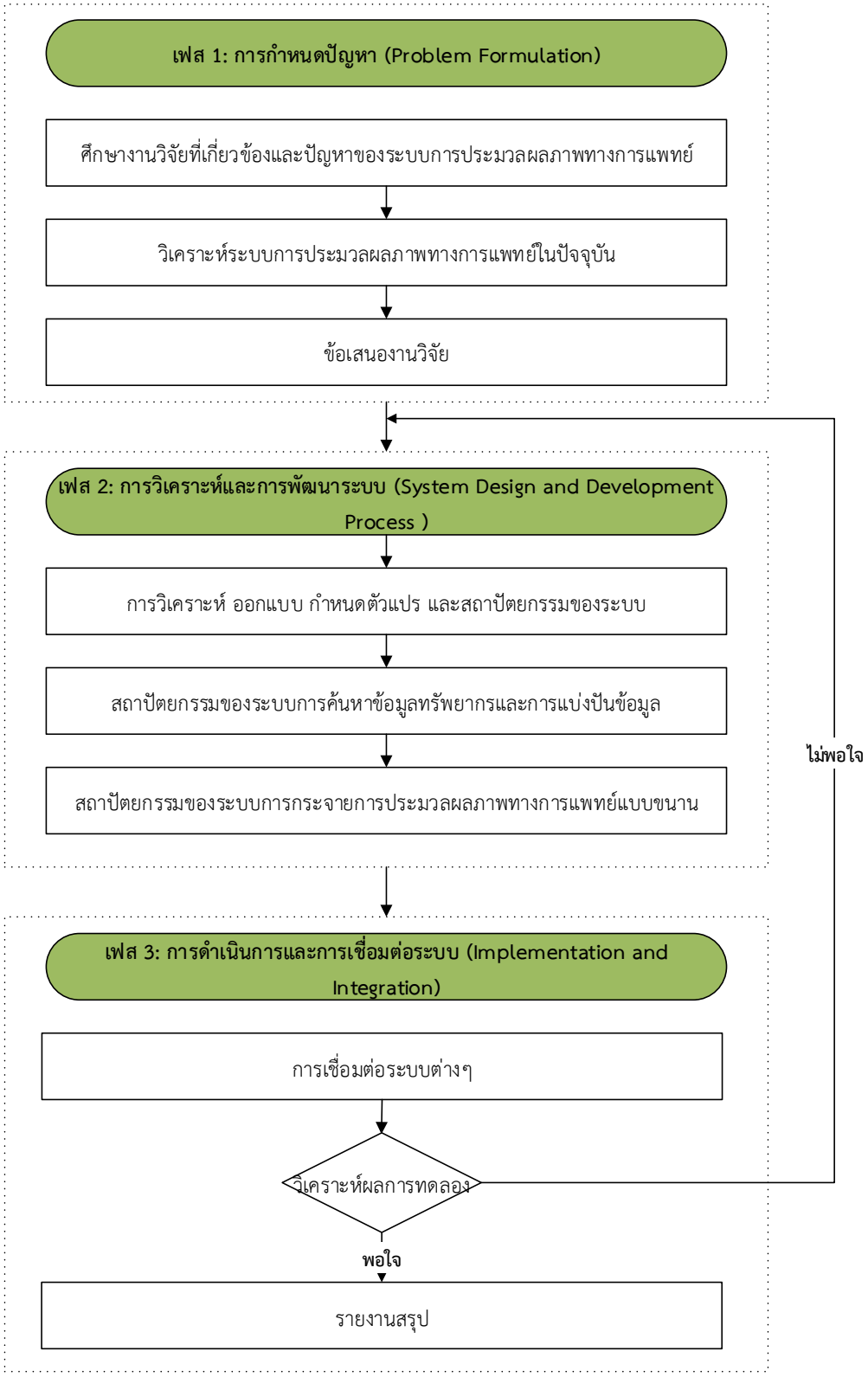
สรุป

เนื้อหาในบทนี้ได้อธิบายในรายละเอียดของมาตรฐานของการจัดการภาพทางแพทย์ เทคโนโลยีกริด (Grid Technology) สถาปัตยกรรมของระบบกริด โกลบัสทูลคิท (Globus Toolkit) วีทีเคทูลคิท (VTK: Visualization Toolkit) การสร้างภาพวิชวลไลเซชัน (Visualization) และกรอบแนวคิดในการวิจัย ตามลำดับ นอกจากนี้ในส่วนของงานวิจัย เทคนิคต่างๆ ที่ได้คิดค้นและพัฒนามาโดยกลุ่มนักวิจัยในอดีตได้ทำการสรุปออกมาในประเด็นต่างๆ ที่น่าสนใจ เช่น เทคนิคที่ใช้ในการพัฒนาผลการทดลอง ตลอดจนปัญหาของการประมวลผลภาพทางการแพทย์ เป็นต้น

บทที่ 3
วิธีดำเนินการวิจัย

กรอบในการวิจัยจะมุ่งเน้นไปที่การพัฒนาโปรแกรมต้นแบบระบบการประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์บนระบบกริด โดยโปรแกรมสามารถกระจายงานการประมวลผลไปยังเครื่องต่าง ๆ ในระบบเครือข่ายได้อย่างมีประสิทธิภาพ ซึ่งการกระจายงานประมวลผลไปยังเครื่องต่าง ๆ ในระบบ โปรแกรมจะทำการคำนวณปริมาณข้อมูลภาพที่เหมาะสมเพื่อที่จะส่งไปยังเครื่องต่าง ๆ โดยใช้ปริมาณการใช้งานของซีพียูเป็นเงื่อนไขในการคำนวณ ผลที่ได้จะทราบปริมาณข้อมูลที่ต้องส่งผ่านโดย GridFTP ไปยังเครื่องต่าง ๆ ในระบบ จากหลักการนี้เป็นการทำ Load Balancing ในข้อมูลด้วย ในบทนี้ได้แสดงกรอบการทำวิจัย (Research Operational Framework) เริ่มจากเฟสแรก การกำหนดปัญหา (Problem Formulation) เป็นการศึกษาปัญหาและวิธีการแก้ปัญหาของการประมวลผลภาพทางการแพทย์จากงานวิจัยในอดีต ทำการวิเคราะห์ระบบประมวลผลภาพทางการแพทย์ในปัจจุบัน เฟส 2 การวิเคราะห์และการพัฒนาระบบ (System Design and Development Process), เป็นเฟสที่จะทำการพัฒนาโปรแกรมระบบโดยการศึกษาวิธีการแก้ปัญหาในอดีตจากเฟสที่ 1 ซึ่งเฟสนี้เริ่มจากวิเคราะห์ ออกแบบโปรแกรมระบบ การประกาศค่าตัวแปรที่สำคัญและออกแบบสถาปัตยกรรมของระบบ ประกอบด้วย สถาปัตยกรรมของระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูล สถาปัตยกรรมของระบบการกระจายการประมวลผลภาพทางการแพทย์แบบขนาน และเฟสที่ 3 การดำเนินการและการเชื่อมต่อระบบ (Implementation and Integration), เฟสนี้จะเริ่มจากการเชื่อมต่อระบบต่าง ๆ ที่ได้ออกแบบและพัฒนาในเฟสที่ 2 หลังจากนั้นทำการทดสอบโปรแกรมระบบโดยการเปรียบเทียบระหว่างจำนวนเครื่องที่ประมวลผลจากน้อยไปหามาก และเทียบกับการกระจายงานเพื่อประมวลผลภาพแบบพื้นฐาน ผลที่ได้จะทำการวิเคราะห์ ถ้าผลการทดลองเป็นที่พอใจก็ทำรายงานสรุป แต่ถ้าผลไม่เป็นที่พอใจให้กลับไปเริ่มเฟสที่ 2 ใหม่เพื่อวิเคราะห์ และออกแบบสถาปัตยกรรม ตลอดจนกระบวนการต่าง ๆ ใหม่

รูปแบบการดำเนินการวิจัย (Research Operational Framework)



ภาพที่ 3.1 รูปแบบการดำเนินการวิจัย

กระบวนการดำเนินการวิจัยดังภาพที่ 3.1 ประกอบด้วย 3 เฟสหลัก ซึ่งแต่ละเฟสสามารถอธิบายในรายละเอียดดังนี้

1. การกำหนดปัญหา (Problem Formulation)

เป็นการศึกษาปัญหาและวิธีการแก้ปัญหาจากงานวิจัยในอดีต ทำการวิเคราะห์ปัญหาและวิธีการแก้ปัญหาเพื่อออกแบบระบบ ตลอดจนการพัฒนาระบบในปัจจุบันเพื่อบรรลุวัตถุประสงค์ในการใช้งานทรัพยากรในระบบกริดอย่างมีประสิทธิภาพและประมวลผลได้อย่างรวดเร็ว นอกจากมีการกำหนดตัวแปรที่เกี่ยวข้องกับการประมวลผลภาพทางการแพทย์และระบบกริด ซึ่งตัวแปรเหล่านี้มีรายละเอียดดังนี้

1.1 ลักษณะของงานที่ประมวลผลในระบบกริด

งานที่ประมวลผลในระบบกริดคือเซตของงาน (J) ที่ประกอบด้วยงานหลายงานที่มีลักษณะแตกต่างกัน กล่าวคือ

$$J=\{J_1,J_2,J_3,\ldots,J_j\} \quad (3.1)$$

โดยที่ J_j เป็นงานที่มีอิสระจากงานย่อยอื่นๆ (Independent Job) (Braun and others. 1998: 330), (Braun and others. 2001: 810) หรืองานที่มีความสัมพันธ์กับงานย่อยอื่น ๆ (Dependent Job) (Yu, & Buyya. 2005: 44; Dhodhi, Ahmad, Yatama, & Ahmad. 2002: 1338) นั้นคือ

$$J_j=\{T^j_1,T^j_2,T^j_3,\ldots,T^j_t\} \quad (3.2)$$

1.2 ทรัพยากรในระบบกริด

เซตของทรัพยากรที่ประกอบด้วยหลายๆ ส่วนที่สามารถนำไปแบ่งปันเพื่อการประมวลผลได้ นั้นคือ

$$R=\{R_1,R_2,R_3,\ldots,R_r\} \quad (3.3)$$

และในแต่ละทรัพยากรประกอบด้วยเซตของเครื่องคอมพิวเตอร์ที่มีลักษณะแตกต่างกัน นั้นคือ

$$R=\{M_1,M_2,M_3,\ldots,M_m\} \quad (3.4)$$

ดังนั้นสามารถนิยามได้ว่าระบบกริดคือ เวกเตอร์ของทรัพยากรทั้งหมด

$$\vec{G}=[\vec{R}_1\ \vec{R}_2\ \vec{R}_3\ \ldots\ \vec{R}_r] \quad (3.5)$$

และเวกเตอร์ของทรัพยากรประกอบด้วยจำนวนเครื่องคอมพิวเตอร์ที่สามารถประมวลผลได้

$$\vec{R}=\begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ . \\ . \\ . \\ M_m \end{bmatrix} \tag{3.6}$$

ดังนั้นในการประมวลผลในระบบกริด เป็นดังนี้

$$\text{การประมวลผล} = \begin{bmatrix} J_1 & J_2 & J_3 \ldots & J_j \end{bmatrix} . \begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ . \\ . \\ . \\ M_m \end{bmatrix} \tag{3.7}$$

จากสมการ (3.7) เป็นการประมวลบนระบบกริดซึ่งจำนวนงานจะมีมากกว่าจำนวนเครื่องคอมพิวเตอร์ในระบบกริด ($J > M$) ดังนั้นในการประมวลผลบนระบบกริดจึงมีความจำเป็นต้องพัฒนาระบบการค้นหาข้อมูลของทรัพยากรและการแบ่งปันข้อมูล และระบบการกระจายการประมวลผลบนระบบกริดให้มีประสิทธิภาพทั้งนี้เนื่องจากทรัพยากรในระบบ เช่น การใช้งานซีพียูของคอมพิวเตอร์และการใช้งานเครือข่าย มีการเปลี่ยนแปลงตลอดเวลา

2. การวิเคราะห์และการพัฒนาระบบ (System Design and Development Process)

เป็นการวิเคราะห์ การออกแบบ การกำหนดค่าตัวแปร และสถาปัตยกรรมทั้งหมดของระบบ ซึ่งสถาปัตยกรรมระบบสามารถแบ่งออกได้ดังนี้ ระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูล และระบบการกระจายการประมวลผลภาพทางการแพทย์แบบขนาน

2.1 การวิเคราะห์ ออกแบบ กำหนดค่าตัวแปร และสถาปัตยกรรมของระบบ

ปัญหาในการประมวลผลบนระบบกริดให้มีประสิทธิภาพประการหนึ่งคือการมอบหมายงานเพื่อประมวลผลบนระบบกริดซึ่งมีความจำเป็นต้องทราบลักษณะงาน (Job Model) และลักษณะของเครื่องคอมพิวเตอร์ที่ประมวลผล (Machine Model) ตลอดจนวัตถุประสงค์ของการประเมินประสิทธิภาพระบบ ซึ่งกลุ่มนักวิจัย เช่น (Quan, Altmann, & Yang. 2008: 274; Yu & Shi. 2007) ได้ทดสอบประสิทธิภาพของระบบด้วยการประเมินระยะเวลาการประมวลผล (Execution time) และระยะเวลาที่แล้วเสร็จ (Finish time) เป็นต้น ในการประเมินประสิทธิภาพของระบบกริด

มีความจำเป็นต้องทราบลักษณะงาน (Job Model) และลักษณะของเครื่องคอมพิวเตอร์ที่ประมวลผล (Machine Model) มีรายละเอียดดังนี้

1) ลักษณะงาน (Job Model)

โมเดลที่ใช้อธิบายคุณสมบัติของงานอยู่ในรูปแบบของ *‹JobID, Description, HostID›* ซึ่งในวิจัยนี้ ลักษณะงานมีดังนี้

- *JobID* หมายเลขของงานที่ส่งไปประมวลผลบนระบบกริด
- *HostID* หมายเลขเครื่องคอมพิวเตอร์บนระบบกริดที่สามารถประมวลผลได้
- *Description* รายละเอียดของลักษณะงาน ได้แก่ สถาปัตยกรรมของซีพียูที่ใช้ในการประมวลผลเป็นแบบไหน ประมวลผลบนโอเอสอะไร ตลอดจนข้อมูลที่ใช้ในการประมวลผลมีปริมาณเท่าไหร่

2) ลักษณะของเครื่องคอมพิวเตอร์ที่ประมวลผล (Machine Model)

โมเลดที่ใช้อธิบายคุณลักษณะของเครื่องคอมพิวเตอร์ที่อยู่ในรูปแบบของ *‹MachineID, Description›* ในวิจัยนี้ ลักษณะของเครื่องคอมพิวเตอร์มีดังนี้

- *MachineID* หมายเลขเครื่องคอมพิวเตอร์บนระบบกริดที่สามารถประมวลผลได้
- *Description* รายละเอียดของเครื่องคอมพิวเตอร์ในระบบกริด ได้แก่ จำนวนซีพียู (CPU) ความเร็วของซีพียู (CPU Speed) การใช้งานซีพียู (CPU Workload)

3) การประเมินประสิทธิภาพของระบบกริด (Performance evaluation)

งานวิจัยนี้พิจารณาประสิทธิภาพของระบบดังนี้

- ระยะเวลาการประมวลผล (Execution time) ประกอบด้วยระยะเวลาของการประมวลผลของแต่ละงาน ($T_{i,comp}$) และระยะเวลาของการส่งข้อมูลผ่านระบบเครือข่าย ($T_{i,comm}$)

$$ET_i = T_{i,comp} + T_{i,comm} \tag{3.8}$$

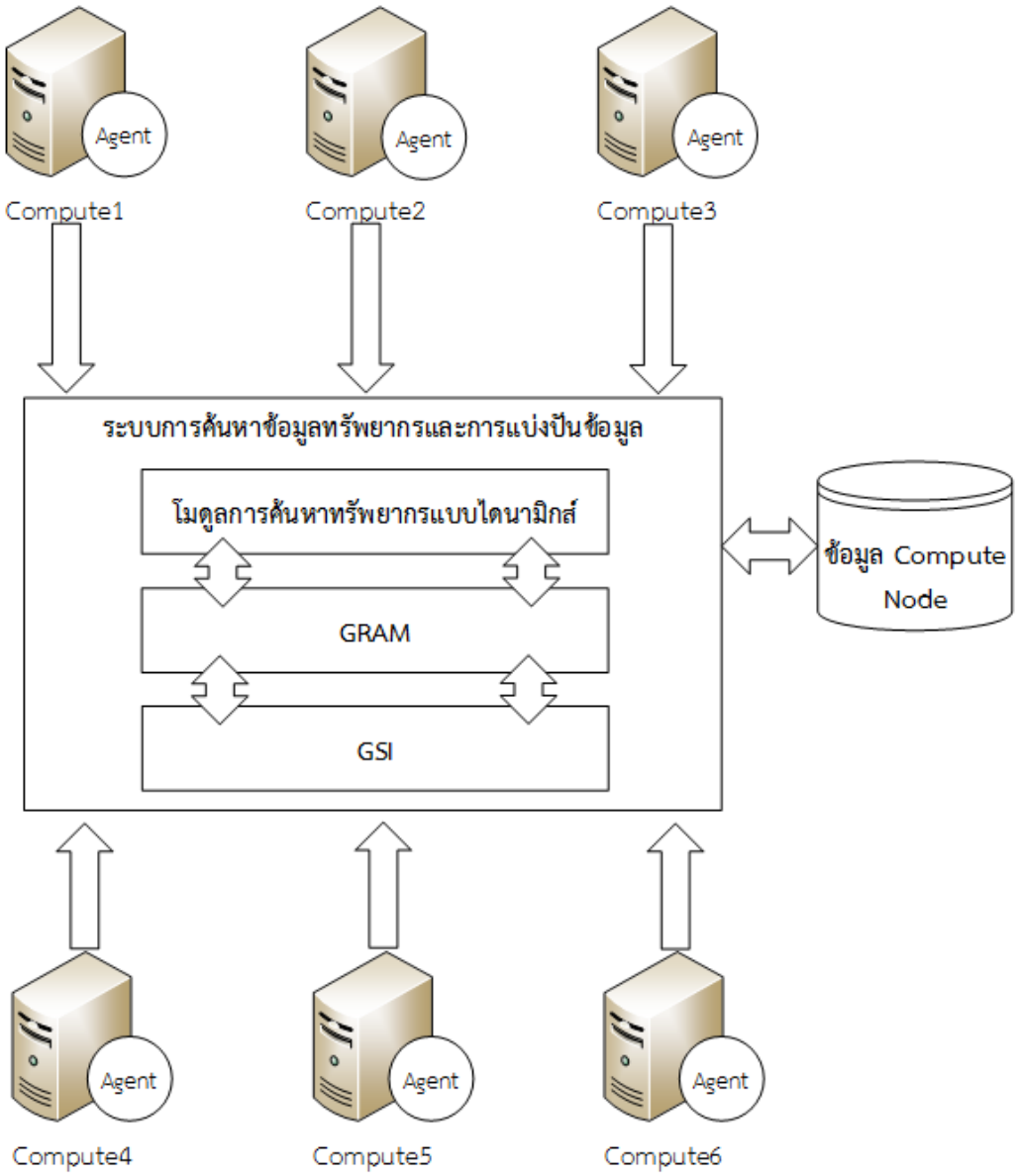
- ระยะเวลาการประมวลของแต่ละเครื่อง (Finish Time)

$$FT = T_{i,comp} \tag{3.9}$$

2.2 สถาปัตยกรรมระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูล

ระบบกริดเป็นระบบที่มีการรวบรวมเอาทรัพยากรจากองค์กรต่างๆเข้าด้วยกันและสามารถแบ่งปันทรัพยากรเพื่อการประมวลผลให้กับหน่วยงานอื่น ๆ ได้ ซึ่งการประมวลผลให้มีประสิทธิภาพมีความจำเป็นต้องมีข้อมูลของเครื่องคอมพิวเตอร์ในระบบกริด ดังนั้นจึงได้ทำการวิจัยพัฒนาระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูลเพื่อให้ระบบดังกล่าวเป็นแหล่งแลกเปลี่ยนข้อมูลทรัพยากรของระบบกริด ได้แก่ จำนวน CPU และความเร็วของเครื่องคอมพิวเตอร์ ตลอดจนการใช้งาน CPU (CPU Workload) ซึ่งจะทำให้ระบบกริดมีข้อมูลต่าง ๆ และสถานการณ์การใช้งาน ณ ปัจจุบันของ

เครื่องคอมพิวเตอร์ การออกแบบและสถาปัตยกรรมของระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูลมีรายละเอียดดังภาพที่ 3.2

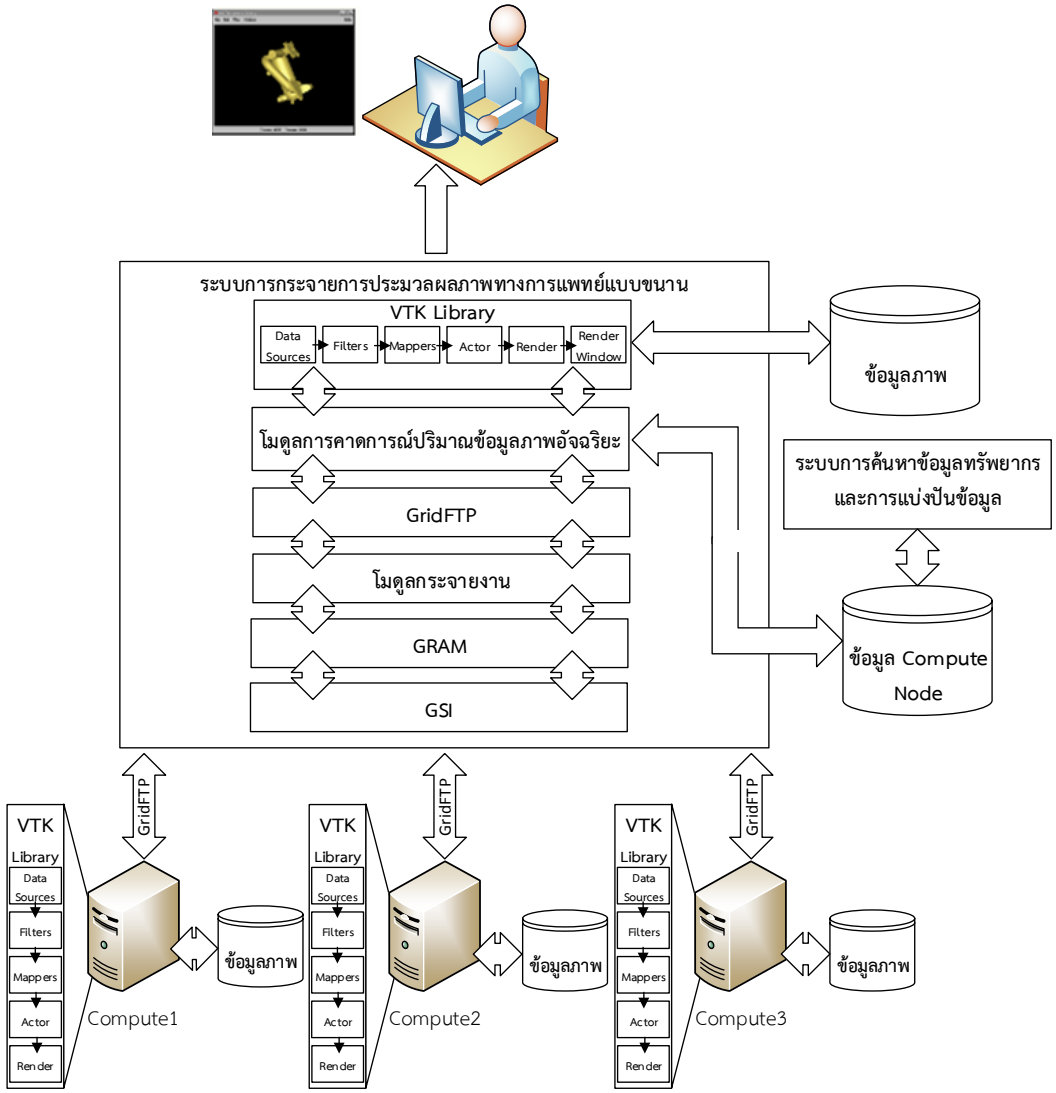


ภาพที่ 3.2 สถาปัตยกรรมของระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูล

ระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูลเริ่มต้นจากเครื่องคอมพิวเตอร์รีจีสเข้ามาในระบบกริด ซึ่งต้องผ่านกระบวนการตรวจสอบด้านความมั่นคง และยืนยันตัวตนของระบบกริด (GSI) หลังจากนั้นข้อมูลของเครื่องคอมพิวเตอร์ที่รีจีสเข้ามาในระบบ เช่น จำนวน CPU ความเร็ว และปริมาณการใช้งาน CPU (CPU Workload) จะถูกบันทึกโดยโมดูลการค้นหาทรัพยากรแบบไดนามิกส์ที่ทำงานร่วมกับ GRAM ในออกแบบและการพัฒนาระบบดังกล่าวได้พัฒนา Agent ที่ฝังตัวอยู่ตามเครื่องคอมพิวเตอร์ในระบบกริดเพื่อทำหน้าที่ส่งข้อมูลกลับมายังเครื่องแม่ข่ายของระบบกริดโดยข้อมูลทั้งหมดได้ถูกรวบรวมไว้ในฐานข้อมูล

2.3 สถาปัตยกรรมระบบการกระจายการประมวลผลภาพทางการแพทย์แบบขนาน

การประมวลผลในระบบกริดให้มีประสิทธิภาพนอกจากข้อมูลต่าง ๆ ของเครื่องคอมพิวเตอร์ที่ใช้ในการประมวลผลแล้ว การมอบหมายงานให้ประมวลผลในแต่ละเครื่องคอมพิวเตอร์อย่างมีประสิทธิภาพเป็นอีกประเด็นที่ต้องให้ความสำคัญ ดังนั้นจึงมีการออกแบบและสถาปัตยกรรมของระบบการกระจายการประมวลผลภาพทางการแพทย์แบบขนานมีรายละเอียดดังภาพที่ 3.3



ภาพที่ 3.3 สถาปัตยกรรมของระบบการกระจายการประมวลผลภาพทางการแพทย์แบบขนาน

ในการประมวลผลภาพทางการแพทย์ เมื่อเครื่องคอมพิวเตอร์ได้ลงทะเบียนเข้ามาในระบบกริดและข้อมูลต่าง ๆ ของเครื่องได้ถูกรวบรวมโดยระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูลแล้ว ข้อมูลของเครื่องคอมพิวเตอร์ต่าง ๆ เหล่านี้ได้ถูกนำไปใช้ในการวิเคราะห์โดยโมดูลการคาดการณ์ปริมาณข้อมูลภาพอัจฉริยะ ในการคำนวณหาปริมาณข้อมูลภาพเพื่อส่งไปประมวลผลในแต่ละเครื่องคอมพิวเตอร์ โมดูลดังกล่าวได้คำนวณตามสูตรดังนี้

$$Vol_Image = \sum_{i=0}^{n} (Speed_i * CPU_i * Load_i * Node) \quad (3.10)$$

โดยที่ *Vol_Image* คือปริมาณภาพถ่ายทางการแพทย์

$Speed_i$ คือความเร็ว CPU ของเครื่องที่ *i* หน่วยเป็น GHz

CPU_i คือจำนวน CPU ของเครื่องที่ *i*

$Load_i$ คือความสามารถคงเหลือในการประมวลผลของ CPU หน่วยเป็น ร้อยละ

เมื่อได้ปริมาณภาพที่ต้องส่งไปยังเครื่องต่าง ๆ แล้วระบบจะทำการส่งภาพเหล่านั้นไปโดยใช้บริการ GridFTP หลังจากนั้นระบบจะทำการสั่งให้แต่ละเครื่องทำการประมวลผลภาพถ่ายทางการแพทย์โดยใช้ VTK Library ที่ติดตั้งในแต่ละเครื่องและโปรแกรมที่พัฒนาขึ้น ซึ่งโปรแกรมที่พัฒนาขึ้นได้ทำการแบ่งปริมาณภาพเพื่อประมวลผลตามจำนวน CPU ของแต่ละเครื่อง ดังนี้

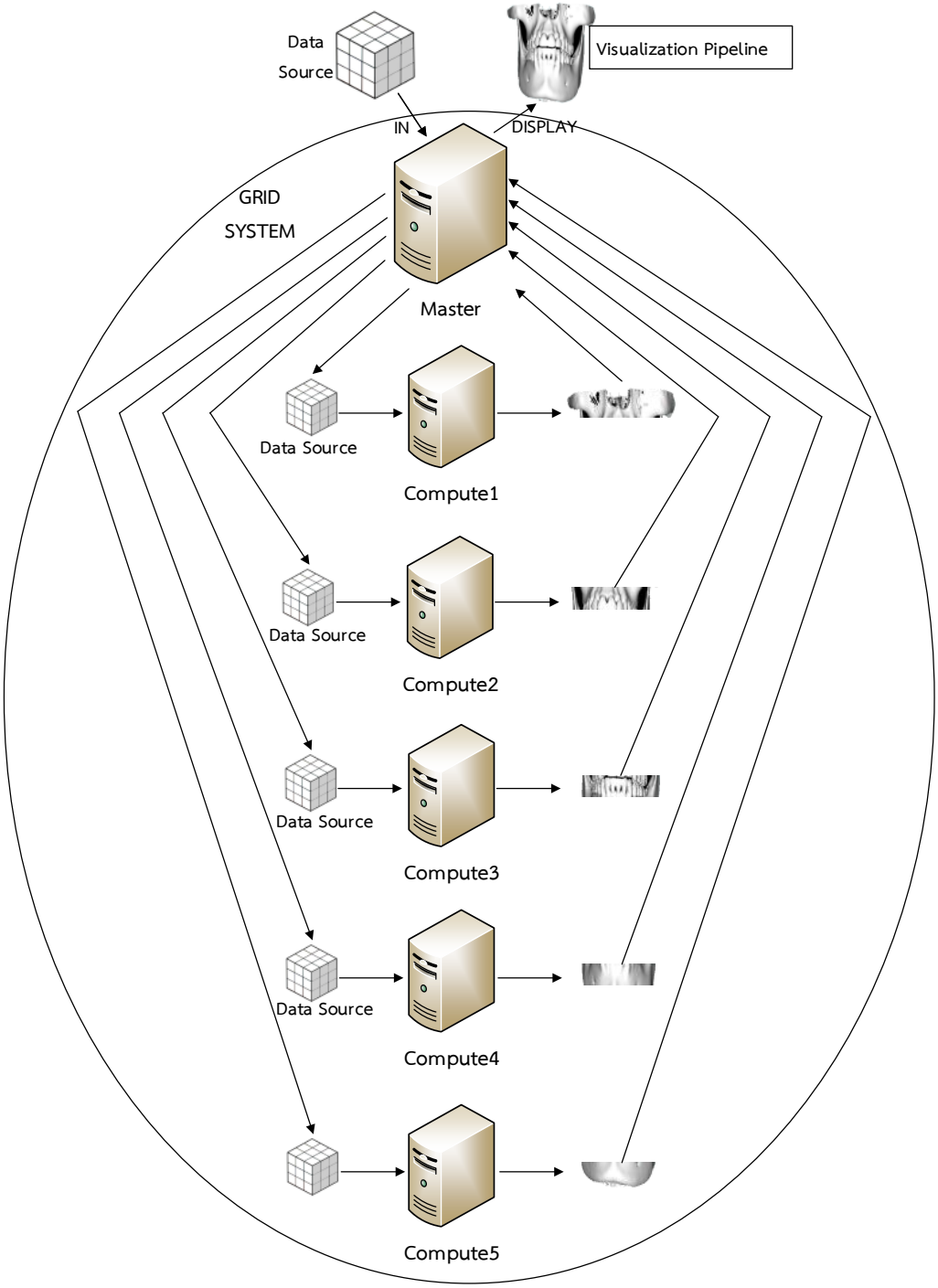
$$CPU_i = images/จำนวนCPU_m \quad (3.11)$$

โดยที่ $CPU_{i,m}$ คือCPUหมายเลข *i* ของเครื่องที่ *m*

images คือปริมาณภาพ

$จำนวนCPU_m$ คือจำนวนCPU ของเครื่องที่ *m*

เมื่อแต่ละเครื่องได้ประมวลผลภาพเสร็จแล้ว ภาพที่ผ่านกระบวนการสร้างภาพเสมือนในรูปแบบไปป์ไลน์ (Visualization Pipeline) ได้ส่งกลับไปยังเครื่องแม่ข่ายของระบบกริดผ่านบริการ GridFTP หลังจากภาพที่ถูกรวบรวมโดยเครื่องแม่ข่ายของระบบกริด ภาพเหล่านี้จะถูกประมวลผลเป็นภาพที่สมบูรณ์ที่สามารถนำไปใช้ในการวินิจฉัยโรคของแพทย์ผู้เชี่ยวชาญได้อย่างมีประสิทธิภาพ ดังรายละเอียดในภาพที่ 3.4



ภาพที่ 3.4 การกระจายการประมวลผลภาพทางการแพทย์แบบขนาน

3. การดำเนินการและการเชื่อมต่อระบบ (Implementation and Integration)

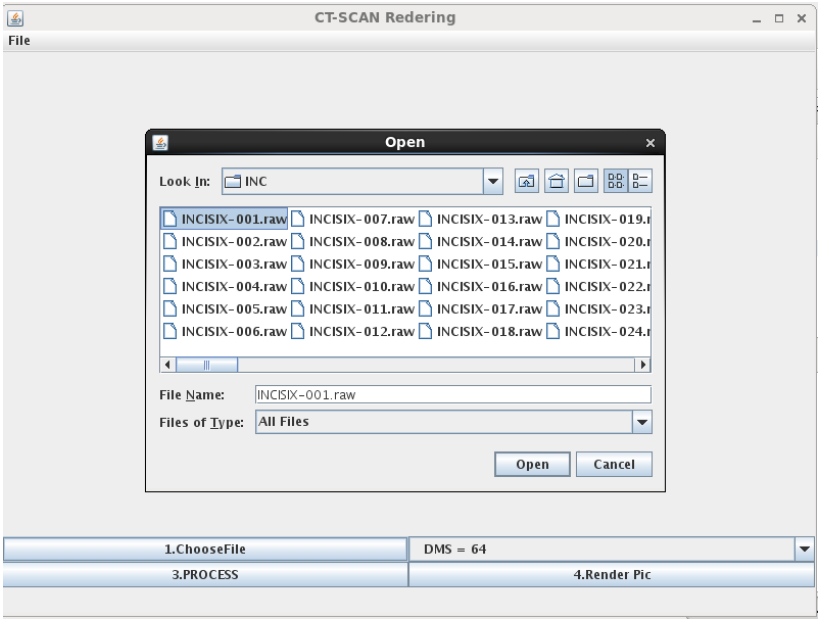
เฟสนี้จะทำการพัฒนาโปรแกรมตามสถาปัตยกรรมที่ได้ออกแบบในเฟสที่ 2 โดยจะทำการเชื่อมต่อระบบต่าง ๆ ซึ่งรายละเอียดของ Flowchart ของระบบ ดังภาพที่ 3.7 ประกอบด้วย 4 ส่วน ดังนี้ 1) ผู้ใช้

2) ระบบการกระจายการประมวลผลภาพทางการแพทย์แบบขนาน

3) ระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูล

4) ทรัพยากรระบบกริด

ในการทำงาน ระบบทำการสร้างระบบกริดก่อนโดยเครื่องคอมพิวเตอร์รีจีสเข้ามาซึ่งต้องผ่านกระบวนการตรวจสอบด้านความมั่นคง และยืนยันตัวตนของระบบกริด (GSI) เมื่อเครื่องคอมพิวเตอร์เป็นสมาชิกของระบบกริดแล้วข้อมูลต่าง ๆ ของเครื่องคอมพิวเตอร์เช่น จำนวนหน่วยประมวลผลกลาง (CPU) ความเร็วของ CPU ตลอดจนสถานะการใช้งาน CPU (CPU Workload) จะถูกบันทึกลงฐานข้อมูลที่บริการโดย GRAM เมื่อผู้ใช้ต้องการประมวลผลภาพทางการแพทย์ ผู้ใช้จะทำการเลือกภาพที่ต้องการประมวลผล ดังภาพที่ 3.5 เป็นกลุ่มภาพ 3D Dental Scans ประกอบด้วย 167 ภาพย่อย (slices) ที่มีความละเอียด 256 x 256 x 256 พิกเซล (pixel) และมีขนาดไฟล์ทั้ง 58 MB หลังจากนั้น

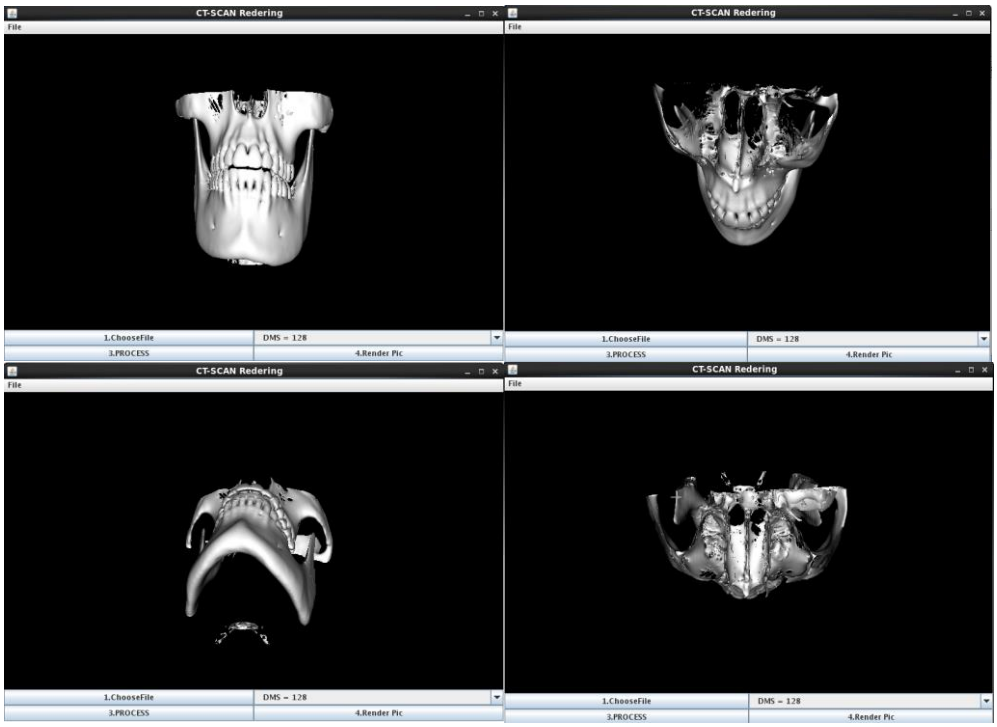


ภาพที่ 3.5 3D Dental Scans Data source

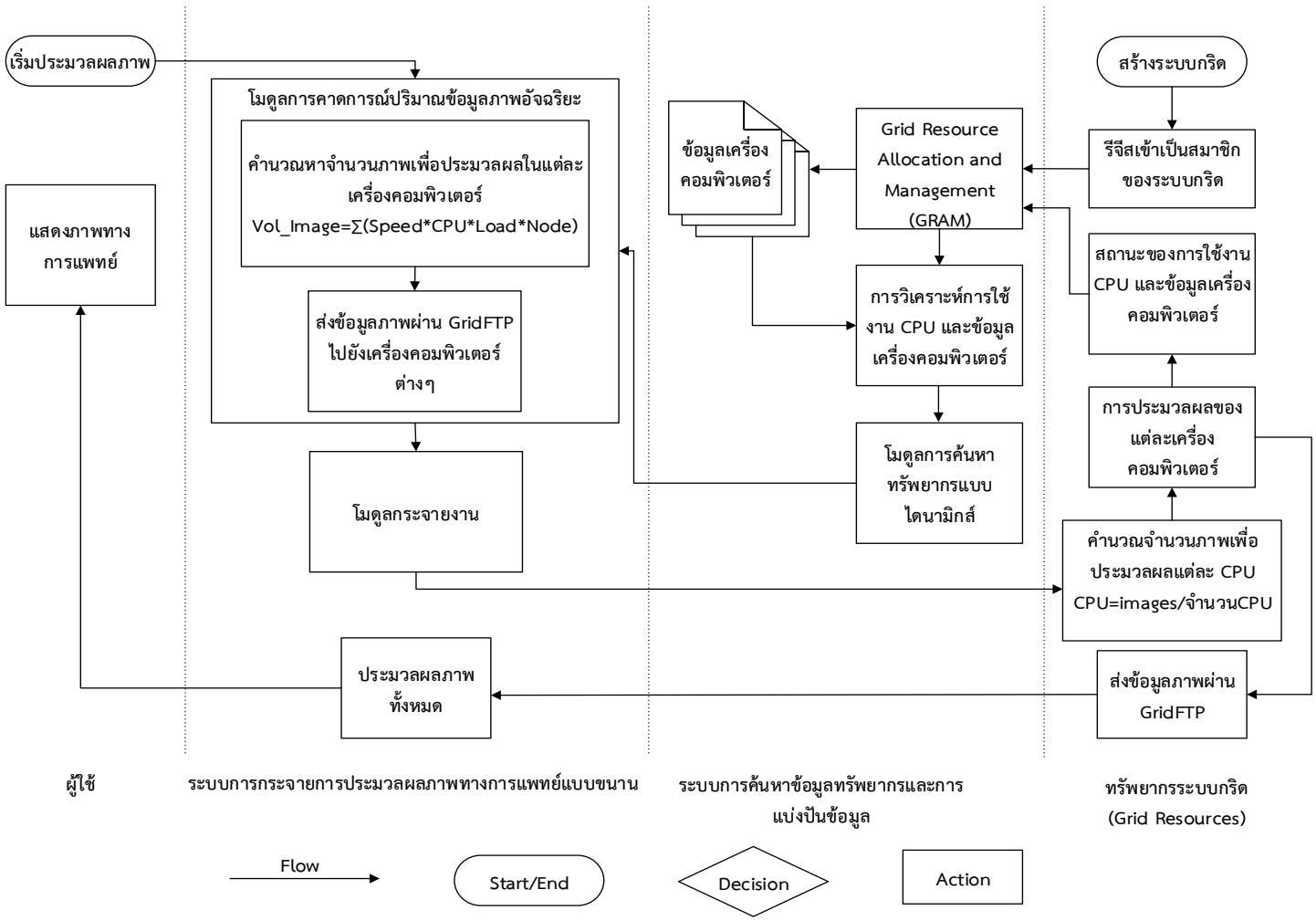
ข้อมูลภาพเหล่านี้จะถูกนำไปคำนวณในโมดูลการคาดการณ์ปริมาณข้อมูลภาพอัจฉริยะ ดังสูตรที่ (3.10) ในส่วนของระบบการกระจายการประมวลผลภาพทางการแพทย์แบบขนานเพื่อหาปริมาณ

ข้อมูลภาพเพื่อส่งไปประมวลผลในแต่ละเครื่องคอมพิวเตอร์ในระบบกริด การหาปริมาณข้อมูลภาพดังสูตร (3.10) ระบบจะใช้ข้อมูลของเครื่องคอมพิวเตอร์ที่มีการค้นหาจากโมดูลการค้นหาทรัพยากรแบบไดนามิกส์ของระบบการค้นหาข้อมูลทรัพยากรและการแบ่งปันข้อมูล เมื่อได้ปริมาณข้อมูลภาพแล้วเครื่องแม่ข่ายของระบบกริดจะทำการส่งข้อมูลภาพเหล่านั้นตามปริมาณที่ได้คำนวณไว้แล้วผ่านบริการ GridFTP ไปยังเครื่องคอมพิวเตอร์ เมื่อข้อมูลภาพที่ส่งไปถูกบันทึกยังเครื่องคอมพิวเตอร์ต่าง ๆ เรียบร้อยแล้วระบบจะทำการประมวลภาพในแต่ละเครื่องโดยใช้ข้อมูลภาพที่ได้รับจากเครื่องแม่ข่ายระบบกริดในการประมวลผลของแต่ละเครื่อง ซึ่งโปรแกรมระบบจะทำการแบ่งปริมาณภาพเพื่อประมวลผลตามจำนวน CPU ของเครื่องคอมพิวเตอร์ ดังสูตรที่ (3.11) หลังจากการประมวลผลเสร็จแล้วแต่ละเครื่องจะทำการส่งข้อมูลภาพที่ประมวลผลเรียบร้อยแล้วไปยังเครื่องแม่ข่ายระบบกริดผ่านบริการ GridFTP ระบบการกระจายการประมวลผลภาพทางการแพทย์แบบขนาน ทำการรวบรวมข้อมูลการประมวลผลจากเครื่องในระบบกริดแล้วทำการประมวลผลภาพทางการแพทย์ที่สมบูรณ์ที่สามารถนำไปใช้ในการวินิจฉัยโรคของแพทย์ผู้เชี่ยวชาญได้อย่างมีประสิทธิภาพต่อไป ดังภาพที่ 3.6

นอกจากการพัฒนาระบบและการเชื่อมต่อระบบต่างๆ แล้วในเฟสนี้ยังทำการทดลองเพื่อหาวิธีการที่มีประสิทธิภาพ แล้วทำการวิเคราะห์ผลการทดลองถ้าหากผลการทดลองเป็นที่พึงพอใจนักวิจัยทำการสรุปรายงานต่าง ๆ ต่อไป



ภาพที่ 3.6 Visualization Pipeline of 3D Dental Scans



ภาพที่ 3.7 Flowchart ไดอะแกรม ของระบบทั้งหมด

ประชากรและการสุ่มกลุ่มตัวอย่าง

ในการทดลองได้ใช้ Data Set เป็น 3D Dental Scans ประกอบด้วย 167 ภาพย่อย (slices) ที่มีความละเอียด 256 x 256 x 256 พิกเซล (pixel) และมีขนาดไฟล์ทั้งสิ้น 58 MB

เครื่องมือในการวิจัยและการตรวจสอบคุณภาพเครื่องมือ

คุณลักษณะของ Software

1. CentOS version 6.5
2. Globus Toolkit version 6
3. VTK Toolkit Library version 5.8
4. Ganglia 3.5.12
5. Java version 1.8

คุณลักษณะของ Hardware

1. เครื่องคอมพิวเตอร์ลูกข่าย 9 เครื่อง ประกอบด้วย
   - จำนวน CPU ระหว่าง 1 – 3 cpus
   - ความเร็วของ CPU 2.4 GHz
   - หน่วยความจำ (Ram) 2 GB
2. เครื่องคอมพิวเตอร์แม่ข่าย 1 เครื่อง ประกอบด้วย
   - จำนวน CPU ระหว่าง 4 cpus
   - ความเร็วของ CPU 2.4 GHz
   - หน่วยความจำ (Ram) 2 GB
3. ระบบเครือข่าย เครื่องคอมพิวเตอร์ทุกเครื่องเชื่อมต่อด้วยสาย LAN ความเร็ว 1000MB Ethernet

การเก็บรวบรวมข้อมูล

ทำการทดลองและประเมินประสิทธิภาพของโปรแกรมระบบที่พัฒนาขึ้นเทียบกับโปรแกรมที่ใช้โมดูลมาตรฐานทั่วไปเช่น การแบ่งจำนวนภาพเท่า ๆ กันเพื่อส่งไปประมวลผลที่เครื่องลูกข่าย ซึ่งการทดลองได้ทำการทดลองซ้ำจำนวน 10 ครั้ง เพื่อหาค่าเฉลี่ยของการทดลองทั้งหมด

การวิเคราะห์ข้อมูล

สรุปผลการทดลองเพื่อเปรียบเทียบระหว่างโปรแกรมระบบที่พัฒนาขึ้นเทียบกับโปรแกรมที่ใช้โมดูลมาตราฐานทั่วไป การประเมินประสิทธิภาพของโปรแกรมระบบเพื่อสร้างภาพเสมือนในรูปแบบไปป์ไลน์ (Visualization Pipeline) ประกอบด้วย ระยะเวลาการประมวลผล (Execution time) ดังสูตรที่ (3.8) และระยะเวลาการประมวลของแต่ละเครื่อง (Finish Time) ดังสูตรที่ (3.9)

สรุป

เนื้อหาที่สำคัญของบทนี้ได้อธิบายในรายละเอียดต่าง ๆ ของรูปแบบการดำเนินการวิจัย (Research Operational Framework) ประกอบด้วย 3 เฟส ดังนี้

1. การกำหนดปัญหา (Problem Formulation)
2. การวิเคราะห์และการพัฒนาระบบ (System Design and Development Process)
3. การดำเนินการและการเชื่อมต่อระบบ (Implementation and Integration)

นอกจากนี้ยังอธิบายในรายละเอียดของ ประชากรและการสุ่มกลุ่มตัวอย่าง เครื่องมือในการวิจัยและการตรวจสอบคุณภาพเครื่องมือ การเก็บรวบรวมข้อมูล ตลอดจนการวิเคราะห์ข้อมูล ซึ่งทั้งหมดนี้เป็นกระบวนการในการทำวิจัย

บทที่ 4
ผลการวิจัย

จากการศึกษาค้นคว้าวิจัยการประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์บนระบบกริด ซึ่งสถาปัตยกรรมของระบบและโปรแกรมได้มีการอธิบายรายละเอียดในบทที่ 3 ในการออกแบบการทดลองเพื่อทดสอบประสิทธิภาพการทำงานของโปรแกรมระบบ โดยพิจารณาตัวชี้วัดดังนี้

1. ระยะเวลาการประมวลผล (Execution time)
2. ระยะเวลาการประมวลของแต่ละเครื่อง (Finish Time)

ในการวัดประสิทธิภาพของระบบในการประมวลผลภาพ ซึ่งตัววัดประสิทธิภาพของการประมวลผลที่รวดเร็ว (High Performance) คือ ระยะเวลาการประมวลผล ขณะที่ระยะเวลาการประมวลของแต่ละเครื่องเป็นตัวชี้วัดประสิทธิภาพของการกระจายงานไปยังเครื่องต่างๆ (Load Balancing)

การออกแบบการทดลอง (Testbed Implementation)

ในการทดลองเพื่อทดสอบประสิทธิภาพการทำงานของโปรแกรมระบบ ได้ออกแบบการทดลองประกอบด้วยข้อมูลภาพทางการแพทย์และโครงสร้างของระบบกริด ดังมีรายละเอียดในตารางที่ 1 และตารางที่ 2 ดังต่อไปนี้

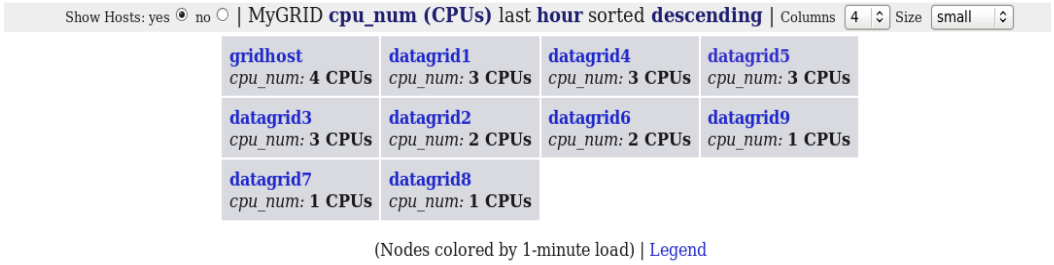
ตารางที่ 4.1 ชุดข้อมูลในการทดสอบ

ชื่อโมเดล	จำนวนของ Polygons	จำนวน slices	Size of Data File
3D Dental Scan	14.62 million	167	58 MB

ตารางที่ 4.2 คุณลักษณะคอมพิวเตอร์

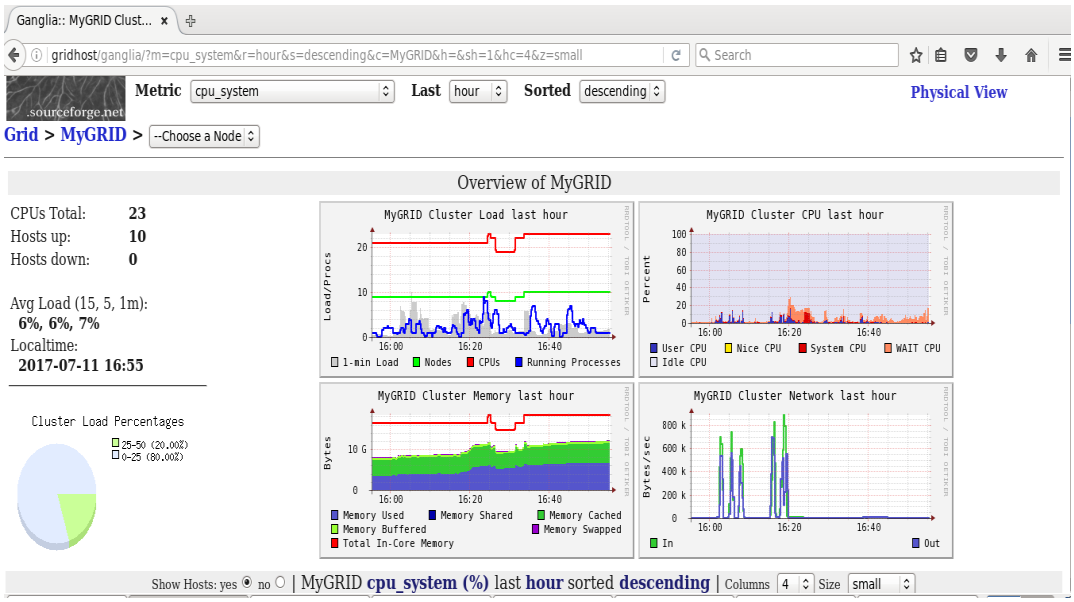
รายการ	จำนวน CPU	ความเร็ว (GHz)	หน่วยความจำ (GB)	ระบบปฏิบัติการ (OS)	จำนวน (เครื่อง)
เครื่องลูกข่าย	1-3	2.4	2	CentOS v.6.5	9
เครื่องแม่ข่าย	4	2.4	2	CentOS v.6.5	1

ระบบเครือข่าย เครื่องคอมพิวเตอร์ทุกเครื่องเชื่อมต่อด้วยสาย LAN ความเร็ว 1000 MB Ethernet การติดตั้งระบบกริดเพื่อทดสอบระบบที่วิจัยและพัฒนาขึ้น มีรายละเอียดในภาพที่ 4.1 และภาพที่ 4.2 โดยภาพที่ 4.1 เป็นภาพที่ประกอบด้วยจำนวนเครื่องคอมพิวเตอร์ที่เป็นสมาชิกของระบบกริดและจำนวน CPU ของแต่ละเครื่องคอมพิวเตอร์ นอกจากนี้ภาพที่ 4.1 ยังแสดงว่าเครื่อง gridhost ที่ประกอบด้วย CPU จำนวน 4 ตัว เป็นเครื่องแม่ข่ายของระบบกริด



ภาพที่ 4.1 จำนวน CPU ของเครื่องคอมพิวเตอร์ที่ใช้ทดลอง

ภาพที่ 4.2 เป็นภาพที่แสดงจำนวนเครื่องคอมพิวเตอร์ทั้งหมดของระบบกริดและจำนวน CPUs ในระบบที่สามารถใช้ประมวลผลได้ นอกจากนี้ภาพที่ 4.2 เป็นการแสดงการใช้งานของ CPUs เพื่อประมวลผล (Work load) การใช้งานหน่วยความจำของระบบทั้งหมด และการใช้งานระบบเครือข่ายของระบบ



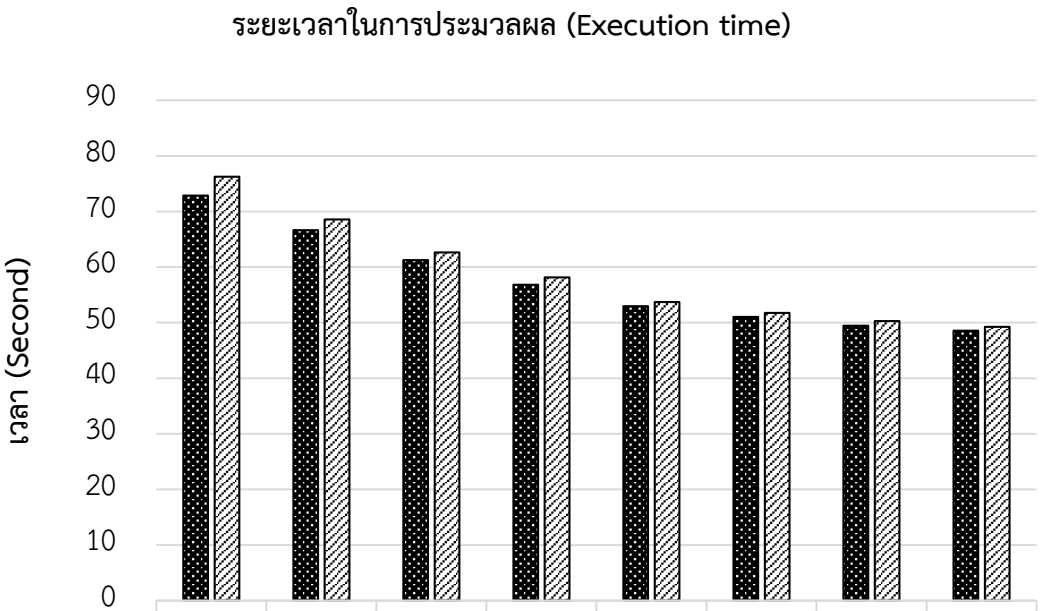
ภาพที่ 4.2 การใช้งานระบบกริดเพื่อประมวลผล และทรัพยากรทั้งหมดของระบบ

ผลการทดลองด้านระยะเวลาการประมวลผล

ในการทดลองการประมวลผลภาพบนระบบกริด ได้ทดลองกับเครื่องคอมพิวเตอร์ตั้งแต่ 3 เครื่องจนถึง 10 เครื่อง จำนวน 10 ครั้งเพื่อหาค่าเฉลี่ย ซึ่งผลการทดลองดังตารางที่ 4.3

ตารางที่ 4.3 ระยะเวลา (Second) ในการประมวลผลภาพทางการแพทย์บนระบบกริดของระบบที่พัฒนาขึ้น

ครั้งที่	จำนวนเครื่อง															
	3		4		5		6		7		8		9		10	
	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual
1	73.67	77.01	65.12	68.21	60.11	62.04	57.43	58.01	53.23	54.41	49.91	50.78	49.54	50.2	48.52	49.01
2	71.12	76.32	67.42	67.51	61.23	62.67	56.98	58.97	52.29	53.21	51.07	51.98	48.65	49.89	48.01	48.89
3	75.34	77.21	67.19	68.45	62.07	63.01	57.97	59.21	52.68	52.99	49.98	52.3	49.02	49.67	48.43	48.91
4	70.84	75.6	68.22	69.65	60.89	61.78	55.23	57.92	53.03	54.03	50.65	51.04	49.45	50.04	47.91	49.51
5	77.91	77.95	66.71	67.91	61.34	62.87	56.01	57.89	52.45	53.11	51.45	51.87	48.9	50.19	48.32	48.87
6	72.04	75.4	65.42	68.69	60.97	62.9	57.58	58.96	54.68	55.65	52.58	52.76	50.03	50.53	49.02	49.33
7	76.1	76.51	66.88	69.03	61.01	63.32	56.82	57.01	52.88	53.5	51.34	51.54	48.78	49.91	48.43	49.21
8	71.76	75.11	67.45	69.56	63.07	63.76	57.01	58.2	53.72	53.89	51.09	52.01	50.33	50.95	48.89	49.35
9	69.94	76.01	65.9	68.9	60.58	61.64	56.08	57.67	52.54	53.56	51.54	52.05	50.02	50.78	49.11	49.55
10	70.03	75.55	66.01	67.77	61.4	62.43	57.09	57.76	52.05	52.8	50.81	51.11	49.71	50.66	48.99	49.9
เฉลี่ย	**72.88**	**76.27**	**66.63**	**68.57**	**61.27**	**62.64**	**56.82**	**58.16**	**52.96**	**53.72**	**51.04**	**51.74**	**49.44**	**50.28**	**48.56**	**49.25**



	3	4	5	6	7	8	9	10
Auto	72.875	66.632	61.267	56.82	52.955	51.042	49.443	48.563
Manual	76.267	68.568	62.642	58.16	53.715	51.744	50.282	49.253

แผนภูมิที่ 4.1 แสดงการเปรียบเทียบระยะเวลาเฉลี่ยการประมวลผลระหว่าง Auto และManual

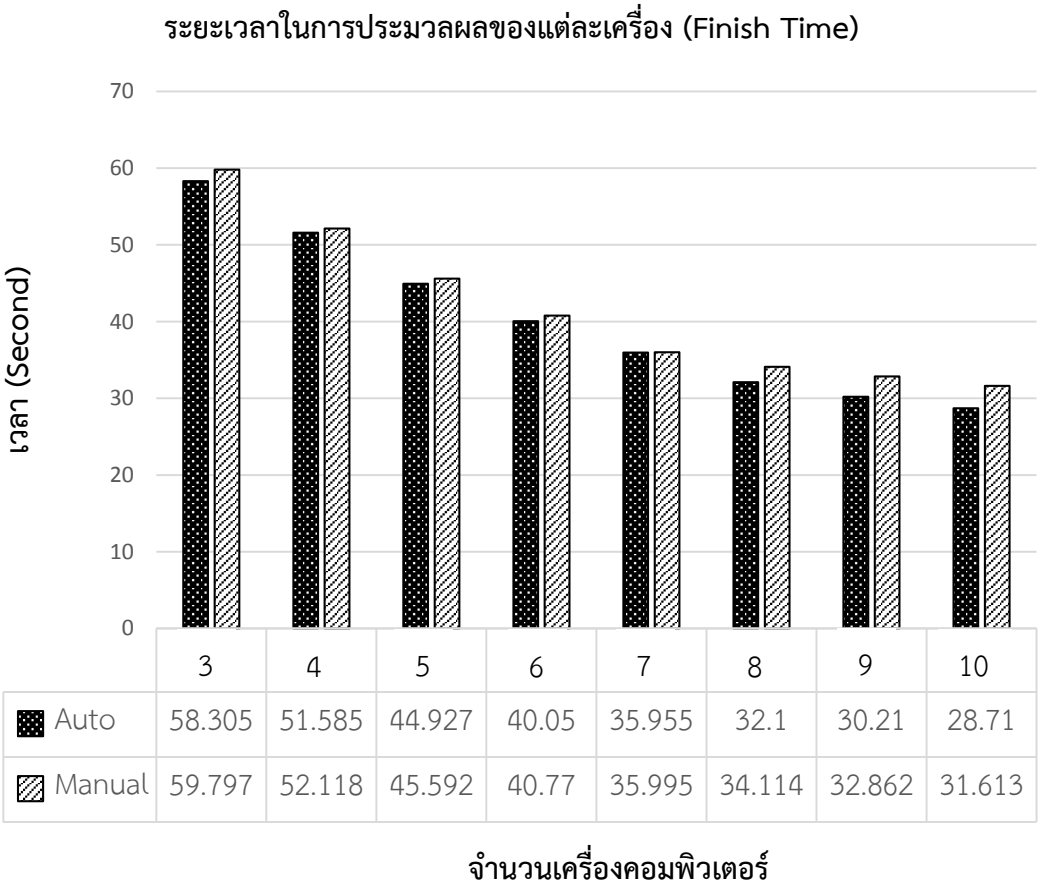
การจากตารางที่ 4.3 สามารถสรุปได้ดังแผนภูมิที่ 4.1 ซึ่งแสดงการเปรียบเทียบระหว่างโปรแกรมระบบที่พัฒนาขึ้น (Auto) และโปรแกรมพื้นฐาน (Manual) ผลการทดลองสรุปได้ว่าระบบที่พัฒนาขึ้นมีประสิทธิภาพในประมวลผลดีกว่าโปรแกรมพื้นฐาน (Manual) และเมื่อเพิ่มจำนวนเครื่องคอมพิวเตอร์ขึ้นมาทดลองครั้งละ 1 เครื่อง ระยะเวลาในการประมวลผลจะลดลงทั้ง 2 แบบ แต่อย่างไรก็ตามระยะเวลาในการประมวลผลทั้ง 2 แบบ ไม่ได้ลดลงอย่างรวดเร็วเหมือนช่วงแรก ๆ ที่เครื่องคอมพิวเตอร์ยังมีไม่มาก ทั้งนี้เพราะถ้าหากเครื่องคอมพิวเตอร์มีจำนวนมากทำให้ระยะเวลาในการส่งข้อมูลไปและกลับต้องเสียเวลาในการรอคิวในระบบเครือข่าย ประกอบกับจำนวนภาพที่ Render ตามเครื่องคอมพิวเตอร์มีจำนวนมากขึ้นทำให้เสียเวลาในการ Render เพื่อให้ได้ภาพสมบูรณ์ที่เครื่องแม่ข่ายของระบบกริด

การทดลองระยะเวลาในการประมวลผลของแต่ละเครื่อง

ในการทดลองการประมวลผลภาพบนระบบกริด ได้ทดลองกับเครื่องคอมพิวเตอร์ตั้งแต่ 3 เครื่องจนถึง 10 เครื่อง จำนวน 10 ครั้งเพื่อหาค่าเฉลี่ย ซึ่งผลการทดลองดังตารางที่ 4.4

ตารางที่ 4.4 ระยะเวลา (Second) ในการประมวลผลภาพทางการแพทย์บนระบบกริดของแต่ละเครื่อง

ครั้งที่	จำนวนเครื่อง															
	3		4		5		6		7		8		9		10	
	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual	Auto	Manual
1	58.67	60.91	49.95	51.41	44.11	44.94	40.93	40.51	36.33	36.61	32.81	33.78	31.84	32.9	31.22	31.31
2	58.02	60.22	52.92	51.41	45.03	45.77	40.08	41.27	35.39	35.71	34.07	34.88	31.15	32.69	30.91	31.09
3	61.84	60.51	51.69	52.15	46.17	46.31	40.87	41.91	35.38	35.29	32.48	35	31.72	32.57	31.13	31.51
4	54.54	58.3	53.42	52.55	44.79	44.88	38.83	40.82	36.23	36.73	33.35	34.14	32.05	32.54	30.71	31.81
5	63.31	61.65	51.91	52.31	44.94	45.37	38.81	40.19	35.25	35.61	34.25	33.87	31.3	32.59	30.82	30.97
6	56.24	59	49.62	52.59	43.77	46.2	39.98	41.16	37.48	37.75	35.28	34.76	32.73	33.03	31.82	32.03
7	61.2	59.31	51.98	52.53	45.71	46.12	40.32	39.91	36.18	35.7	34.14	33.54	31.58	32.21	30.73	31.41
8	57.66	58.51	52.85	52.66	45.97	46.06	40.51	40.8	36.72	35.79	33.69	34.01	32.63	33.75	31.49	31.85
9	54.74	60.11	51.2	51.9	43.68	45.24	39.58	40.37	35.44	35.36	34.64	34.05	32.62	33.48	32.01	32.15
10	56.83	59.45	50.31	51.67	45.1	45.03	40.59	40.76	35.15	35.4	33.11	33.11	32.21	32.86	31.49	32
เฉลี่ย	**58.31**	**59.8**	**51.59**	**52.19**	**44.93**	**45.59**	**40.05**	**40.77**	**35.96**	**36.0**	**33.78**	**34.11**	**31.98**	**32.86**	**31.23**	**31.61**



	3	4	5	6	7	8	9	10
Auto	58.305	51.585	44.927	40.05	35.955	32.1	30.21	28.71
Manual	59.797	52.118	45.592	40.77	35.995	34.114	32.862	31.613

แผนภูมิที่ 4.2 แสดงการเปรียบเทียบระยะเวลาเฉลี่ยการประมวลผลระหว่าง Auto และManual ของเครื่องคอมพิวเตอร์แต่ละเครื่อง

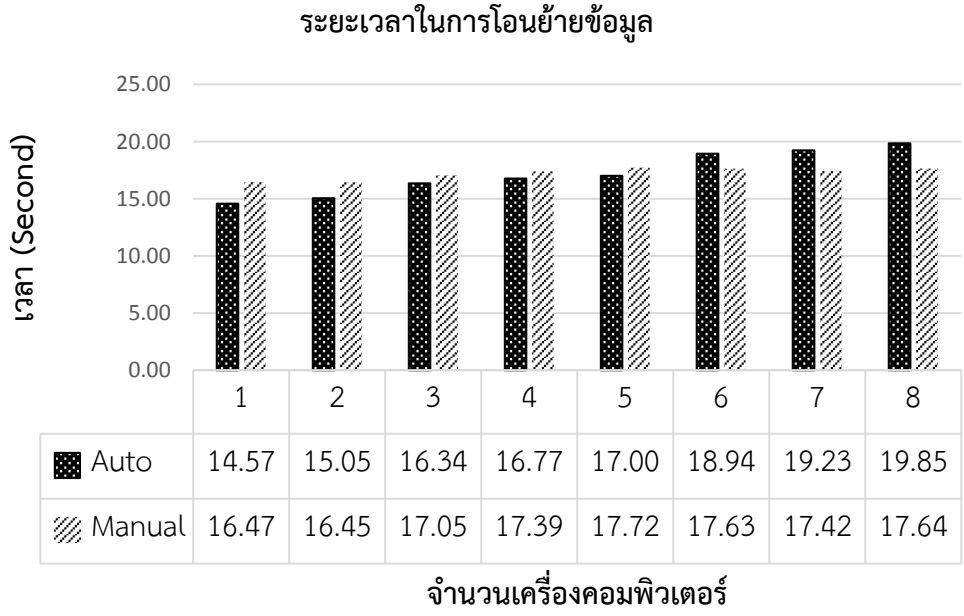
จากตารางที่ 4.4 สามารถสรุปได้ดังแผนภูมิที่ 4.2 ซึ่งแสดงการเปรียบเทียบระหว่างโปรแกรมระบบที่พัฒนาขึ้น (Auto) และโปรแกรมพื้นฐาน (Manual) ผลการทดลองสรุปได้ว่าระบบที่พัฒนาขึ้นมีประสิทธิภาพในการประมวลผลต่อเครื่องคอมพิวเตอร์ได้ดีกว่าโปรแกรมพื้นฐาน (Manual) และเมื่อเพิ่มจำนวนเครื่องคอมพิวเตอร์ขึ้นมาทดลองครั้งละ 1 เครื่อง ระยะเวลาในการประมวลผลจะลดลงทั้ง 2 แบบ อย่างไรก็ตามเมื่อจำนวนเครื่องมากขึ้นโปรแกรมระบบที่พัฒนาขึ้น (Auto) สามารถประมวลผลต่อเครื่องได้ดีกว่าแบบโปรแกรมพื้นฐาน (Manual) ทั้งนี้เพราะระบบสามารถกระจายการประมวลผลไปยังเครื่องต่างๆ ได้อย่างมีประสิทธิภาพ

การทดลองระยะเวลาในการโอนย้ายข้อมูล

ในการทดลองการประมวลผลภาพบนระบบกริด ได้ทดลองกับเครื่องคอมพิวเตอร์ตั้งแต่ 3 เครื่องจนถึง 10 เครื่อง จำนวน 10 ครั้งเพื่อหาค่าเฉลี่ย ซึ่งผลการทดลอง ดังตารางที่ 4.5

ตารางที่ 4.5 ระยะเวลาเฉลี่ยในการโอนย้ายข้อมูล

จำนวนเครื่อง	Auto (Second)	Manual (Second)
3	14.57	16.47
4	15.05	16.45
5	16.34	17.05
6	16.77	17.39
7	17.00	17.72
8	18.94	17.63
9	19.23	17.42
10	19.85	17.64



แผนภูมิที่ 4.3 การเปรียบเทียบระยะเวลาเฉลี่ยการประมวลผลระหว่าง Auto และManual ของการโอนย้ายข้อมูล

จากตารางที่ 4.5 สามารถสรุปได้ดังแผนภูมิที่ 4.3 ซึ่งแสดงการเปรียบเทียบระหว่างโปรแกรมระบบที่พัฒนาขึ้น (Auto) และโปรแกรมพื้นฐาน (Manual) ผลการทดลองสรุปได้ว่าระบบที่พัฒนาขึ้นมีประสิทธิภาพในการโอนย้ายข้อมูลได้ดีกว่าในกรณีที่มีจำนวนเครื่องคอมพิวเตอร์ไม่มากเมื่อเทียบกับโปรแกรมพื้นฐาน (Manual) แต่เมื่อจำนวนเครื่องเพิ่มมากขึ้นการโอนย้ายข้อมูลทำได้ไม่ดีพอเมื่อเทียบกับโปรแกรมพื้นฐาน (Manual) ทั้งนี้เพราะในการออกแบบโปรแกรม นักวิจัยเน้นในการประมวลผลที่เครื่องคอมพิวเตอร์แต่ละเครื่องให้มีประสิทธิภาพ ดังนั้นจึงทำการแบ่งภาพออกตามจำนวน CPU ในการประมวลผล เพื่อให้ใช้ทรัพยากรอย่างมีประสิทธิภาพและการประมวลผลที่รวดเร็วยิ่งขึ้น ผลที่ตามมาคือมีจำนวนภาพ Render เกิดขึ้นเป็นจำนวนมากซึ่งมีผลทำให้ภาพเหล่านั้นต้องเข้าคิวรอในระบบเครือข่ายของระบบเป็นผลทำให้ระยะเวลาในการโอนย้ายข้อมูลทำได้ไม่ดีพอ เมื่อเทียบกับโปรแกรมพื้นฐาน (Manual)

สรุป

จากผลการทดลองแสดงการเปรียบเทียบระหว่างโปรแกรมระบบที่พัฒนาขึ้น (Auto) และโปรแกรมพื้นฐาน (Manual) พบว่า โปรแกรมที่พัฒนาขึ้น สามารถประมวลผลภาพทางการแพทย์ได้อย่างมีประสิทธิภาพในด้านของเวลาที่ประมวลผลและสามารถกระจายการประมวลผลไปยังเครื่องคอมพิวเตอร์ลูกข่ายอย่างมีประสิทธิภาพในด้านของเวลาเช่นกัน อย่างไรก็ตามเมื่อเปรียบเทียบในด้านของการโอนย้ายข้อมูลระหว่างเครื่องคอมพิวเตอร์ในระบบที่ออกแบบมา พบว่า เมื่อจำนวนเครื่องคอมพิวเตอร์มีจำนวนเพิ่มขึ้น การโอนย้ายข้อมูลยังทำได้ไม่ดีพอ ดังเหตุผลที่กล่าวไว้แล้วในสรุปผลการทดลองเรื่องการโอนย้ายข้อมูล

บทที่ 5
สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ

สรุปผลการวิจัย

จากผลการดำเนินงานวิจัย สรุปได้จากการเปรียบเทียบค่าระยะเวลาที่ใช้ในการประมวลผลภาพทางการแพทย์พบว่าการประมวลผลโดยโปรแกรมระบบที่พัฒนาขึ้น (Auto) ใช้เวลาในการประมวลผลน้อยกว่าโปรแกรมพื้นฐาน (Manual) แต่เมื่อเพิ่มจำนวนเครื่องคอมพิวเตอร์มากขึ้นระยะเวลาที่ประมวลผลระหว่างทั้ง 2 แบบ มีระยะเวลาไม่ต่างกันมากเหมือนตอนแรก เช่น ในการทดลองเมื่อใช้เครื่องคอมพิวเตอร์ประมวลผล 3 เครื่อง กับเครื่องคอมพิวเตอร์ 10 เครื่อง พบว่าระยะเวลาที่โปรแกรมระบบที่พัฒนาขึ้น (Auto) ประมวลผลได้เร็วกว่าโปรแกรมพื้นฐาน (Manual) เป็น 3.39 วินาที กับ 0.69 วินาที เป็นต้น ทั้งนี้เป็นเพราะการรอคิวในระบบเครือข่ายและการออกแบบโปรแกรมระบบที่พัฒนาขึ้น (Auto) ที่แบ่งจำนวนข้อมูลภาพตามจำนวน CPUs ของแต่ละเครื่องคอมพิวเตอร์ ทำให้ได้จำนวนภาพ Render ที่ต้องโอนย้ายข้อมูลเป็นจำนวนมาก

การประมวลผลข้อมูลภาพทางการแพทย์ของแต่ละเครื่องคอมพิวเตอร์ในระบบ พบว่า โปรแกรมระบบที่พัฒนาขึ้น (Auto) ใช้เวลาในการประมวลผลของแต่ละเครื่องคอมพิวเตอร์น้อยกว่าโปรแกรมพื้นฐาน (Manual) โดยเฉพาะเมื่อมีจำนวนเครื่องคอมพิวเตอร์มากขึ้น ทั้งนี้เพราะระบบทำการตรวจสอบสถานะการใช้งาน CPU (Workload) ก่อนที่จะกระจายการประมวลผลไปตามเครื่องคอมพิวเตอร์ต่าง ๆ สำหรับการโอนย้ายข้อมูลพบว่าเมื่อเครื่องคอมพิวเตอร์มีจำนวนมากขึ้นการโอนย้ายข้อมูลโดยโปรแกรมระบบที่พัฒนาขึ้น (Auto) ทำได้ไม่ดีพอเมื่อเทียบกับโปรแกรมพื้นฐาน (Manual) ดังนั้นการรอคิวเพื่อโอนย้ายข้อมูลในคิวของระบบเครือข่าย และการโอนย้ายข้อมูลของ ภาพ Render ที่มีจำนวนมากมีผลต่อประสิทธิภาพโดยรวมเป็นอย่างมาก

อภิปรายผล

จากสมมติฐานก่อนทำการวิจัยที่มีอยู่ว่า การประมวลผลภาพทางการแพทย์ที่มีความละเอียดสูงมีความจำเป็นต้องใช้เครื่องที่มีสมรรถนะสูงและต้องใช้เทคนิคในการแบ่งการประมวลผลไปยังหน่วยประมวลผลต่างๆ ที่อยู่ในพื้นที่เดียวกัน เช่น คลัสเตอร์ ดังนั้นทำให้มีแนวคิดว่า การนำความสามารถในการประมวลผลภาพทางการแพทย์มาทำการกระจายไปยังหน่วยประมวลผลจำนวนมากที่มีการทำงานอิสระต่อกัน ที่อยู่ต่างพื้นที่และเชื่อมต่อกันด้วยระบบเครือข่ายราคาถูก เช่น อินเตอร์เน็ต จะช่วยให้เพิ่มประสิทธิภาพของการประมวลผลในด้านเวลาและลดค่าใช้จ่ายในการลงทุน

ด้านอุปกรณ์ อย่างไรก็ตามการเพิ่มประสิทธิภาพของการประมวลผลโดยการลดระยะเวลาขึ้นอยู่กับปัจจัยที่สำคัญ 2 ประการคือ เทคนิคที่ช่วยการกระจายการประมวลผลไปยังหน่วยประมวลผลต่างๆ และการโอนย้ายข้อมูลภายในระบบเครือข่าย จากสมมติฐานดังกล่าว นำไปสู่การวิจัยและพัฒนาโปรแกรมประยุกต์การประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์บนระบบกริด โดยใช้โปรแกรมโอเพนซอร์ส (Open Source) เป็นเครื่องมือในการพัฒนาโปรแกรมต้นแบบ ทั้งนี้เพื่อไม่ให้เกี่ยวข้องกับเรื่องลิขสิทธิ์ และสามารถวิจัยและพัฒนาความสามารถของโปรแกรมประยุกต์ต่อไปในอนาคตได้

จากสมมติฐานและแนวความคิดในการพัฒนาโปรแกรมประยุกต์ที่กระจายการประมวลผลไปยังหน่วยประมวลผลต่างๆ ที่อยู่ต่างพื้นที่กัน ทำให้เกิดอุปสรรคในการวิจัยหลายด้าน เช่น

1. การติดตั้งโปรแกรมระบบกริด (Globus Toolkit) เป็นโปรแกรมที่มีขั้นตอนในการติดตั้งและการทำงานที่ซับซ้อน ทำให้เสียเวลาในการศึกษาเรียนรู้เป็นเวลานาน อย่างไรก็ตามมีความจำเป็นต้องติดตั้งโปรแกรมดังกล่าวเพื่อให้สนับสนุนการทำงานบนระบบกริดได้อย่างมีประสิทธิภาพ
2. การพัฒนาโปรแกรมต้นแบบที่ต้องใช้เทคนิคของการกระจายงานไปยังหน่วยประมวลผลต่าง ๆ บนระบบกริดควบคู่กับโปรแกรม VTK Library ที่สนับสนุนการสร้างภาพเสมือนในรูปแบบไปป์ไลน์ (Visualization Pipeline) ต้องใช้เวลาในการศึกษาและวิจัยเป็นเวลา
3. ข้อมูลภาพที่ใช้ในการทดลองโปรแกรมต้นแบบต้องใช้เวลาในการหาเนื่องจากเป็นข้อมูลภาพของผู้ป่วยดังนั้น โรงพยาบาลจะพยายามปกปิดเป็นความลับและไม่เผยแพร่สู่สาธารณะ
4. การพัฒนาโปรแกรมต้นแบบบนระบกริดมีความจำเป็นต้องสร้างสิ่งแวดล้อมให้เหมือนกับระบบดังกล่าวซึ่งต้องอาศัยอุปกรณ์ ทั้งเครื่องคอมพิวเตอร์และอุปกรณ์เครือข่ายต่างๆ ซึ่งมีผลทำให้มีต้นทุนในการสร้างโปรแกรมต้นแบบที่ราคาสูง

อย่างไรก็ตามจากการศึกษาวิจัยการประมวลผลภาพแบบกระจายของข้อมูลทางการแพทย์บนระบบกริด โดยการพัฒนาโปรแกรมประยุกต์ต้นแบบเป็นสิ่งที่คุ้มค่าทั้งนี้เนื่องจากผลการทดลองสามารถสรุปได้ว่าโปรแกรมต้นแบบทำการประมวลผลได้อย่างมีประสิทธิภาพ นอกจากนี้การสร้างโปรแกรมต้นแบบยังสามารถลดการพึ่งพิงเทคโนโลยีจากต่างประเทศ และพัฒนาต่อยอดไปสู่ภาคอุตสาหกรรมในอนาคตได้

ข้อเสนอแนะในการนำผลการวิจัยไปใช้

เนื่องจากเป็นโปรแกรมระบบที่ต้องใช้ระบบเครือข่ายในการแลกเปลี่ยนข้อมูลระหว่างเครื่องคอมพิวเตอร์ ดังนั้นการนำโปรแกรมต้นแบบดังกล่าวไปใช้ต้องพิจารณาดังต่อไปนี้

1. ระบบเครือข่ายของหน่วยงานต้องมีความเสถียรสูงและมีความเร็วสูงทั้งนี้เพื่อเพิ่มประสิทธิภาพของการลดระยะเวลาในการโอนย้ายข้อมูล

2. การแลกเปลี่ยนข้อมูลประกอบด้วย ข้อมูลภาพทางการแพทย์ และข้อมูลสถานะการใช้งานเครื่องคอมพิวเตอร์ ซึ่งมีความจำเป็นต้องแก้ไขระบบความมั่นคงปลอดภัยของเครื่องคอมพิวเตอร์ลูกข่าย ตลอดจนติดตั้งโปรแกรม Agent เพื่อแลกเปลี่ยนข้อมูลกับเครื่องแม่ข่าย ทำให้บางหน่วยงานอาจจะไม่ใช้งาน หรือสร้างอีกระบบหนึ่งแยกออกมาเพื่อใช้งาน แต่ก็ทำให้มีต้นทุนที่เพิ่มขึ้น

3. เนื่องจากโปรแกรมต้นแบบถูกพัฒนาให้สามารถทำงานร่วมกับ VTK Library บนระบบกริด (Globus Toolkit) ซึ่งเป็นการพัฒนาโปรแกรมภายใต้หลักการโอเพนซอร์ส (Open Source) ทำให้มีประเด็นในเรื่องของความเสถียรของโปรแกรม และการสนับสนุนเมื่อโปรแกรมมีปัญหา

ข้อเสนอแนะในการทำวิจัยครั้งต่อไป

จากผลการทดลอง พบว่า ปัญหาที่สำคัญหลายประการที่ทำให้ระยะเวลาในการประมวลผลของระบบไม่รวดเร็วขึ้นเมื่อเพิ่มจำนวนเครื่องคอมพิวเตอร์คือการรอคิวในระบบเครือข่ายระหว่างการโอนย้ายข้อมูลขนาดใหญ่ การสร้างภาพ Render ย่อยๆ ในแต่ละเครื่อง และการรวบรวมภาพ Render ย่อยๆ เพื่อประมวลผลภาพที่สมบูรณ์ที่เครื่องแม่ข่าย ดังนั้นประเด็นที่กล่าวมานี้เป็นสิ่งที่ควรจะต้องศึกษาวิจัยและพัฒนาเพื่อให้การประมวลผลภาพทางการแพทย์แบบกระจายมีประสิทธิภาพเพิ่มขึ้น

บรรณานุกรม

บรรณานุกรมภาษาไทย

เพชรากร หาญพาณิชย์, วัลลภ เหล่าไพบูลย์. (2007). ระบบสื่อสารและการเก็บข้อมูลภาพทางการแพทย์. Srinagarind Medical Journal, 22.

ไชยรส, นายยุทธพงษ์. (2550). การแสดงสิทธิ์อนุมัติแบบ 3 ส่วนสำหรับกริดสาธารณสุขด้านภาพทางการแพทย์. (วิศวกรรมศาสตรมหาบัณฑิต), มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ, มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

“ระบบ PACS” (2017). Cited 2017 Jan 1. Available from: http://med.mahidol.ac.th/sdmc/th/technology/PAC-th

บรรณานุกรมภาษาต่างประเทศ

Aboamama Atahar Ahmed, Muhammad Shafie Abd Latiff, Kamalrulnizam Abu Bakar, and Zainul Ahmad Rajion. (2007). Visualization Pipeline for Medical Datasets on Grid Computing Environment. *The 2007 International Conference on Computational Science and its Applications (ICCSA 2007)*, 26-29 Aug, Kuala Lampur, Malaysia, 567-576.

Aboamama Atahar Ahmed, Muhammad Shafie Abd Latiff, Kamalrulnizam Abu Bakar, and Zainul Ahmad Rajion. (2007). Automatic Visualization Pipeline Formation for Medical Datasets on Grid Computing *Environment. International Journal of Computer, Information, Systems and Control Engineering*, 1(7), 2007-2012.

Aboamama Atahar Ahmed, Muhammad Shafie Abd Latiff, Kamalrulnizam Abu Bakar, Zainul Ahmad Rajion, and Zainul Ahmad Rajion. (2007). Adaptive Resources Selection Framework for Grid Enabled Visualization Pipeline. *IJCSNS International Journal of Computer Science and Network Security*, 7(12). 114-123.

About the Globus Toolkit. (2017). Cited 2017 Jan 1. Available from: http://toolkit.globus.org/toolkit/about.html

Abram, Greg, & Treinish, Lloyd. (1995). An Extended Data-Flow Architecture for Data Analysis and Visualization. *Proceedings of the 6th conference on Visualization '95*, October 29 - November 03, Washington, DC, USA. 263 -270.

Baker, Mark, Buyya, Rajkumar, & Laforenza, Domenico. (2002). Grids and grid technologies for wide-area distributed computing. *Journal of Software Practice & Experience*, 32(15), 1437-1466.

Bardsiri, Marjan Kuchaki Rafsanjani and Amid Khatibi. (2012). A New Heuristic Approach for Scheduling Independent Tasks on Heterogeneous Computing Systems. *International Journal of Machine Learning and Computing*, 2(4), 371-376.

Beeson, B., Dwyer, M., & Thompson, D. (2005). Server-side visualisation of massive datasets. *The First International Conference on e-Science and Grid Computing (e-Science'05)*, 1-1 July. Melbourne, Vic., Australia. 7-548.

Berman, Francine D., Wolski, Rich, Figueira, Silvia, Schopf, Jennifer, & Shao, Gary. (1996). Application-level scheduling on distributed heterogeneous networks. *The Proceedings of the 1996 ACM/IEEE conference on Supercomputing*, 1 Jan, Pittsburgh, Pennsylvania, USA.

Bhaskaran.V, Kamalam.G.K and Murali. (2010). A New Heuristic Approach:Min-mean Algorithm For Scheduling Meta-Tasks On Heterogeneous Computing Systems. *International Journal of Computer Science and Network Security*, 10(1), 24-31.

Braun, T. D., Siegel, H. J., Beck, N., Boloni, L., Maheswaran, M., Reuther, A. I.Bin, Yao. (1998). A taxonomy for describing matching and scheduling heuristics for mixed-machine heterogeneous computing systems. *Proceedings Seventeenth IEEE Symposium on Reliable Distributed Systems*. 20-23 Oct, West Lafayette, IN, USA, USA. 330-335.

Braun, Tracy D, Siegel, Howard Jay, Beck, Noah, B, Ladislau L, Freund, Richard F. (2001). A Comparison of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Heterogeneous Distributed Computing Systems. *Journal of Parallel Distributed Computing*, 61(6), 810-837.

Buyya, Rajkumar, & Murshed, Manzur. (2002). GridSim: a toolkit for the modeling and simulation of distributed resource management and scheduling for Grid computing. *Concurrency and Computation: Practice and Experience*, 14(13-15), 1175-1220.

Dhodhi, Muhammad K., Ahmad, Imtiaz, Yatama, Anwar, & Ahmad, Ishfaq. (2002). An Integrated Technique for Task Matching and Scheduling onto Distributed Heterogeneous Computing Systems. *Journal of Parallel and Distributed Computing*, 62(9), 1338-1361.

Dutra, M., Rodrigues, P. S. S., Giraldi, G. A., & Schulze, B. (2007). Distributed Visualization Using VTK in Grid Environments. *The Seventh IEEE International Symposium on Cluster Computing and the Grid (CCGrid '07)*, 14-17 May. Rio De Janeiro, Brazil. 381-388.

Fewings, A. J., & John, N. W. (2007). Distributed Graphics Pipelines on the Grid. *IEEE Distributed Systems Online*, 8(1), 1-1.

Foster, I., C. Kesselman, J. M. Nick and S. Tuecke. (2002). The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. *Technical report, Globus*, February.

Foster, I., Zhao, Y., Raicu, I., & Lu, S. (2008). Cloud Computing and Grid Computing 360-Degree Compared. *The 2008 Grid Computing Environments Workshop*, 12-16 Nov, Austin, TX, USA. 1-10.

Foster, Ian, Kesselman, Carl, & Tuecke, Steven. (2001). The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *The International Journal of High Performance Computing Applications*, 15(3), 200-222.

Frey, James, Tannenbaum, Todd, Livny, Miron, Foster, Ian, & Tuecke, Steven. (2002). Condor-G: A Computation Management Agent for Multi-Institutional Grids. *Cluster Computing*, 5(3), 237-246.

Grace, R. K., Manimegalai, R., & Kumar, S. S. (2014). Medical Image Retrieval System in Grid Using Hadoop Framework. *The 2014 International Conference on Computational Science and Computational Intelligence*, 10-13 March, Las Vegas, NV, USA. 144-148.

Ian, Foster, & Carl, Kesselman. (1999). The grid: blueprint for a new computing infrastructure. *Morgan Kaufmann Publishers Inc*. San Francisco, CA, USA.

KLUSÁČEK Dalibor, Hana RUDOVÁ. (2008). Improving QoS in Computational Grids through Schedule-based Approach. *The the International Conference on Automated Planning and Scheduling (ICAPS'08)*. Sydney, Australia.

Li, F., Gong, B., & Xing, C. (2008). An Adaptive System of Service-Oriented Visualization on the Grid. 20-22 Aug. *The The Third ChinaGrid Annual Conference (chinagrid 2008)*, 20-22 Aug. Dunhuang, Gansu, China. 229-234.

Liu, Y., & Gao, S. (2009). WSRF-Based Distributed Visualization. *The 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, 18-21 May, Shanghai, China, China. 615-619.

Mark Baker, Rajkumar Buyya, and Domenico Laforenza. (2000). The Grid : International Efforts in Global Computing. *International Conference on Advances in Infrastructure for Electronic Business, Science,and Education on the Internet (SSGRR 2000)*, l`Aquila, Rome, Italy.

Mohammed Bakri Bashir, Muhammad Shafie Abd Latiff, Aboamama Atahar Ahmed, Adil Yousif and Manhal Elfadil Eltayeeb. (2013). Content-based Information Retrieval Techniques Based on Grid Computing: A Review. *IETE TECHNICAL REVIEW*, 30(3), 223-232.

Moreland, K. (2013). A Survey of Visualization Pipelines. *IEEE Transactions on Visualization and Computer Graphics*, 19(3), 367-378.

NEMA. Digital and Communication in Medicine (DICOM). (2008). Cited 2017 Jan 1. Available from: http://www.nema.org/prod/med

Open Grid Services Architecture. (2017). Cited 2017 Jan 1. Available from: http://toolkit.globus.org/ogsa/

Parker, S. G., & Johnson, C. R. (1995, 1995). SCIRun: A Scientific Programming Environment for Computational Steering. *Proceedings of the IEEE/ACM SC95 Conference*. San Diego, CA, USA, USA.

Quan, D. M., Altmann, J., & Yang, L. T. (2008). Mapping Heavy Communication SLA-Based Workflows onto Grid Resources with Parallel Processing Technology.

The 2008 10th IEEE International Conference on High Performance Computing and Communications, 25-27 Sept, Dalian, China. 274-281.

R. Kingsy Grace, B. Priyavarshini, J. Rajasri, H. Rajeshwari. (2012). A Survey On Medical Image Retrieval Implementations In Data Grid. *International Journal of Engineering Research & Technology (IJERT)*, 1(10).

Rodrigues, José Fernando, Jr., Balan, André G. R., Zaina, Luciana A. M., & Traina, Agma J. M. (2015). A Survey on Distributed Visualization Techniques over Clusters of Personal Computers. *INFOCOMP Journal of Computer Science*.

Siriluck Lorpunmanee, Mohd Noor Sap, Abdul Hanan Abdullah, and Chai Chompoo-inwai. (2007). An Ant Colony Optimization for Dynamic Job Scheduling in Grid Environment. *World Academy of Science, Engineering and Technology*, 314-321.

Tang, M., & Qiu, Z. (2009). Parallel visualization of large medical datasets based on computer cluster. *The 9th International Conference on Electronic Measurement & Instruments*, 16-19 August. Beijing, China. 238- 240.

The Visualization Toolkit (VTK). (2017). Cited 2017 Jan 1. Available from: http://www.vtk.org/.

W. Schroeder, K. Martin, and B. Lorensen. (2004). The Visualization Toolkit: An Object Oriented Approach to 3D Graphics (4th ed.): *Kitware Inc.*

Welcome to the Globus Toolkit Homepage. (2017). Cited 2017 Jan 1. Available from: http://toolkit.globus.org/toolkit/

Wiley, George. (2005). Picture Archiving and Communication System (PACs). Cited 2017 Jan 1. Available from: http://www.axisimagingnews.com

Yang, Chao-Tung, Chen, Chiu-Hsiung, & Yang, Ming-Feng. (2010). Implementation of a medical image file accessing system in co-allocation data grids. *Future Generation Computer Systems*, 26(8), 1127-1140.

Yu, Jia, & Buyya, Rajkumar. (2005). A taxonomy of scientific workflow systems for grid computing. *ACM SIGMOD Record*, 34(3), 44-49.

Yu, Z., & Shi, W. (2007). An Adaptive Rescheduling Strategy for Grid Workflow Applications. *IEEE International Parallel and Distributed Processing Symposium*, 26-30 March, Rome, Italy. 1-8.

Zhang, Xiaoyu, Bajaj, Chandrajit, & Blanke, William. (2001). Scalable isosurface visualization of massive datasets on COTS clusters. *The IEEE 2001 symposium on parallel and large-data visualization and graphics*, 23-23 Oct. San Diego, California. 51-59.

ภาคผนวก

ภาคผนวก ก

ขั้นตอนการติดตั้ง Cmake และ VTK Library

ขั้นตอนการติดตั้ง Cmake

ขั้นตอนที่ 1 โหลดไฟล์ cmake-3.0.0 จากนั้นแตกไฟล์ไว้ที่โฟลเดอร์ /usr/local/src

วิธีทำ เข้าเว็บ www.cmake.org/download/ > ดาวโหลดไฟล์ cmake-3.0.0.tar.gz มา > จากนั้นแตกไฟล์ไว้ที่โฟลเดอร์ /usr/local/src

ขั้นตอนที่ 2 ติดตั้ง ncurses-devel

วิธีทำ # yum install ncurses-devel

ขั้นตอนที่ 3 ติดตั้ง cmake-3.0.0

วิธีทำ # ./bootstrap --prefix=/usr/local/cmake-3.0.0
make
make install
ln -s /usr/local/cmake-3.0.0 /usr/local/cmake

ขั้นตอนที่ 4 Export Path ของ Cmake

วิธีทำ # export PATH=/usr/local/cmake/bin:$PATH
export MANPATH=/usr/local/cmake/man:$MANPATH

ขั้นตอนการติดตั้ง VTK library

ขั้นตอนที่ 1 โหลด vtk-5.8.0.tar.gz และ vtkdata-5.8.0.tar.gz มา จากนั้นแตกไฟล์ไว้ที่โฟลเดอร์ /usr/local/src

วิธีทำ เข้าเว็บ http://www.vtk.org/download/ > ดาวโหลดไฟล์ vtk-5.8.0.tar.gz มา > จากนั้นแตกไฟล์ไว้ที่โฟลเดอร์ /usr/local/src

ขั้นตอนที่ 2 Export Path ของ VTK

วิธีทำ #export CMAKE_LIBRARY_PATH=/lib64:/usr/lib64

ขั้นตอนที่ 3 make VTK

วิธีทำ # ccmake .

กด C

CMAKE_INSTALL_PREFIX /usr/local/vtk-5.8.0

VTK_DATA_ROOT /usr/local/vtk-5.8.0/

VTKData VTK_WRAP_PYTHON ON

เลือก JAVA

กด C > C > G

make

mkdir -p /usr/local/vtk-5.8.0/lib/python2.4/site-packages/

make install

ขั้นตอนที่ 4 แตกไฟล์ vtkdata-5.8.0.tar.gz และย้ายไปไว้ที่ /usr/local/vtk/

วิธีทำ # tar xzvf vtkdata-5.8.0.tar.gz

mv VTKData /usr/local/vtk/

ขั้นตอนที่ 5 Export Path VTK

วิธีทำ # export PATH=/usr/local/vtk/bin:$PATH

export LD_LIBRARY_PATH=/usr/local/vtk/lib:$LD_LIBRARY_PATH

export PYTHONPATH=/usr/local/vtk/lib/python2.4/site-packages:$PYTHONPATH

export VTK_DATA_ROOT=/usr/global/vtk/VTKData

mv VTKData /usr/local/vtk/

ขั้นตอนที่ 6 ก็อปปี้ /usr/local/vtk/ ไว้ที่ Desktop

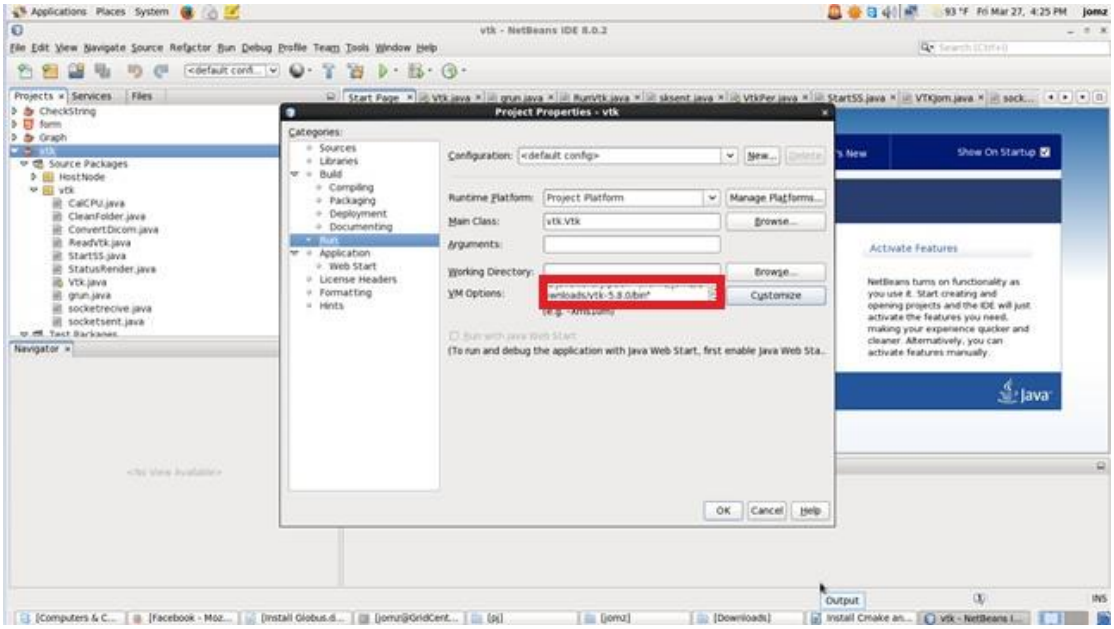
วิธีทำ คลิกขวา > Copy > ไปที่ Desktop > คลิกขวา > วาง

ขั้นตอนการ Set Path และ Libraries ใน Netbeans

ขั้นตอนที่ 1 ใส่ Path VTK

วิธีทำ คลิกขวาที่โปรเจค > Properties > run > ช่อง VM Option ให้ใส่

-Djava.library.path="/home/jomz/Downloads/vtk-5.8.0/bin"

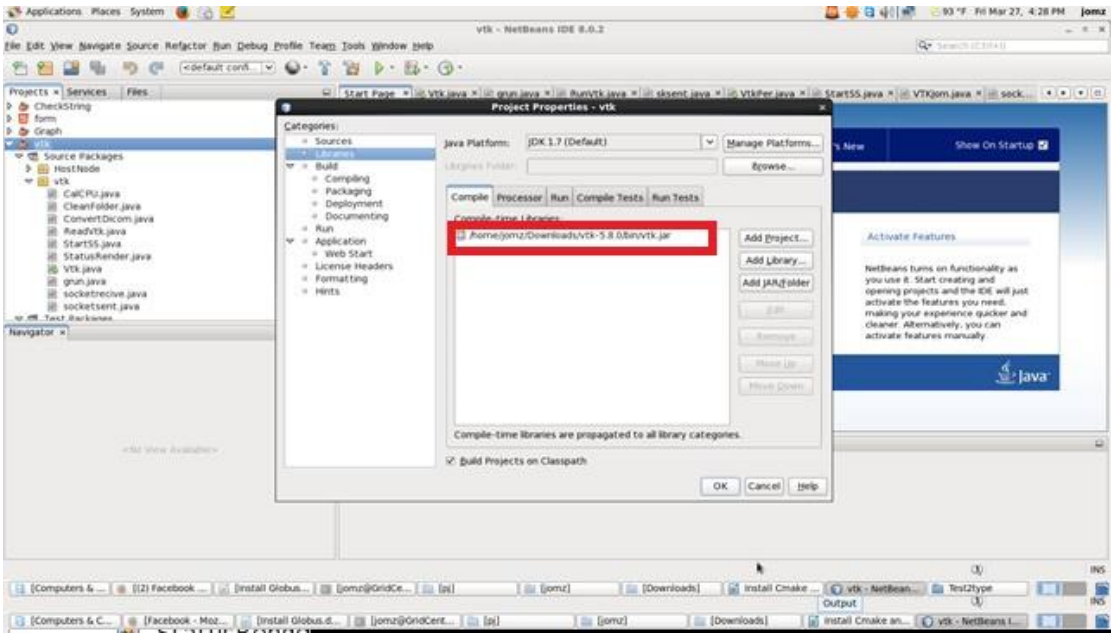


ภาพที่ ก.1 การใส่ Path VTK

ขั้นตอนที่ 2 ใส่ Libraries VTK

วิธีทำ คลิกขวาที่ Project > Properties > Libraries > Add Library

จะอยู่ในโฟลเดอร์ vtk/bin ชื่อไฟล์ vtk.jar



ภาพที่ ก.2 การใส่ Library VTK

ภาคผนวก ข

ขั้นตอนในการติดตั้ง Globus Toolkit

ขั้นตอนการติดตั้ง Globus Toolkit บน CentOS6.5

ขั้นตอนที่ 1 ให้ทำการเพิ่ม IP ของเครื่อง Host และ Node ลงใน /etc/hosts ของทั้งสองเครื่อง โดยเครื่องตัวเองให้ใช้ IP 127.0.0.1

วิธีทำ vi /etc/hosts

127.0.0.1 GridHost.dusit.ac.th

10.123.144.15 GridNode1.dusit.ac.th

10.123.144.16 GridNode2.dusit.ac.th

ขั้นตอนที่ 2 ปิด Firewall

วิธีทำ เปิด terminal > พิม setup > เลือก FireWall Configuration > เอา * ออก

ขั้นตอนที่ 3 ปิด selinux

วิธีทำ เปิด terminal > พิม vi /etc/selinux/config > แก้ดังนี้

This file controls the state of SELinux on the system.

SELINUX= can take one of these three values:

enforcing - SELinux security policy is enforced.

permissive - SELinux prints warnings instead of enforcing.

disabled - No SELinux policy is loaded.

SELINUX=disabled

SELINUXTYPE= can take one of these two values:

targeted - Targeted processes are protected,

mls - Multi Level Security protection.

SELINUXTYPE=targeted

จากนั้น restart เครื่อง

ขั้นตอนที่ 4 ลง voms และ voms-clients

วิธีทำ เวอร์ชั่นที่ใช้ลงคือ

voms-clients-2.0.10-1.el6.x86_64.rpm

voms-2.0.10-1.el6.x86_64.rpm

ขั้นตอนที่ 5 ลง repo

วิธีทำ พิมพ์ rpm –hUv http://www.globus.org/ftppub/gt5/5.2/5.2.5/installers/repo/Globus-5.2.stable-config.fedora-17-1.noarch.rpm

ขั้นตอนที่ 6 สร้าง User

วิธีทำ พิมพ์ useradd <quser> (ในที่นี้จะใช้ quser)
passwd <password>

ขั้นตอนที่ 7 ทำการติดตั้ง Myproxy,GridFTP,GRAM ลงบนเครื่อง Host

วิธีทำ root: yum install globus-gridftp globus-gram5 globus-gsi myproxy myproxy-server myproxy-admin
root: install -o myproxy -m 644 /etc/grid-security/hostcert.pem /etc/grid-security/myproxy/hostcert.pem
root: install -o myproxy -m 600 /etc/grid-security/hostkey.pem /etc/grid-security/myproxy/hostkey.pem
root: vi /etc/myproxy-server.config

ขั้นตอนที่ 8 ทำการคอนฟิก Myproxy

วิธีทำ root:vi /etc/myproxy-server.config
นำเครื่องหมาย # ออกจากหน้าคำด้านล่าง
accepted_credentials "*"
authorized_retrievers "*"
default_retrievers "*"
authorized_renewers "*"
default_renewers "none"
authorized_key_retrievers "*"
default_key_retrievers "none"
trusted_retrievers "*"
default_trusted_retrievers "none"
cert_dir /etc/grid-security/certificates
เสร็จแล้วใช้ root พิมพ์คำสั่ง root:usermod -a -G simpleca myproxy

ขั้นตอนที่ 9 ทำการสร้าง Certificate ให้กับ User

วิธีทำ myproxy@root: PATH=$PATH:/usr/sbin

myproxy@root: myproxy-admin-adduser -c "QuickStart User" -l User

ขั้นตอนที่ 10 ทำการ add grid-mapfile ลงบนเครื่อง Host

วิธีทำ root:grid-mapfile-add-entry –dn "/O=Grid/OU=GlobusTest/OU=simpleCA-hostnameHost /OU=local/CN=QuickStart User" –ln User

ขั้นตอนที่ 11 ทำการเปิด Service เพื่อเตรียมใช้งาน

วิธีทำ root:service myproxy-server start

root:service globus-gridftp-server start

root:service globus-gatekeeper start

ขั้นตอนที่ 12 ทำการติดตั้ง Myproxy , GridFTP , GRAM ลงบนเครื่อง Node

วิธีทำ root: yum install globus-gridftp myproxy globus-gram5

ขั้นตอนที่ 13 ทำการร้องขอไฟล์ จากเครื่อง Host

วิธีทำ myproxy-get-trustroots -b –s hostname(Host)

ขั้นตอนที่ 14 ให้เครื่อง Host ทำการ Add Service ให้เครื่องโหนด

วิธีทำ myproxy@root(Host):myproxy-admin-addservice -c "hostname Node" –l hostname Node

root: myproxy-retrieve -s hostname Host -k hostname Node -l hostname Node

root: myproxy-destroy -s hostname Host -k hostname Node -l hostname Node

ขั้นตอนที่ 15 ทำการ Add grid-mapfile ลงบนเครื่อง Node

วิธีทำ root:grid-mapfile-add-entry -
dn"/O=Grid/OU=GlobusTest/OU=simpleCA-hostname Host
/OU=local/CN=QuickStart User" -ln User

ขั้นตอนที่ 16 ทำการเปิด Service เพื่อเตรียมใช้งาน

วิธีทำ root:service globus-gridftp-server start
root:service globus-gatekeeper start

ขั้นตอนที่ 17 เมื่อทั้งสองเครื่องต้องการใช้ Service ต่างๆบน Globus Toolkit จะต้องทำการ Login Myproxy ก่อน

วิธีทำ user:myproxy-logon -s hostname Host(เครื่อง Host)
user:myproxy-logon -s hostname Host(เครื่อง Node)

ประวัติผู้วิจัย

1. ชื่อ - นามสกุล (ภาษาไทย): นายศิริลักษณ์ หล่อพันธ์มณี
ชื่อ - นามสกุล (ภาษาอังกฤษ): Mr.Siriluck Lorpunmanee

2. เลขหมายบัตรประจำตัวประชาชน: 3 5499 00192 96 6

3. ตำแหน่งปัจจุบัน: ประธานหลักสูตรวิทยาการคอมพิวเตอร์ และ หัวหน้าศูนย์วิจัย High Performance Computing and Networking Centre

4. หน่วยงานและสถานที่อยู่ที่ติดต่อได้สะดวก พร้อมหมายเลขโทรศัพท์ โทรสาร และไปรษณีย์อิเล็กทรอนิกส์
หลักสูตรวิทยาการคอมพิวเตอร์ ศูนย์การเรียนรางน้ำ มหาวิทยาลัยราชภัฏสวนดุสิต เบอร์ 02-642-5596-7, E-mail: siriluck_lor@dusit.ac.th

5. ประวัติการศึกษา
ปี พ.ศ. 2542 ปริญญาโท M.ISM. (Information Systems Management)
สถาบันบัณฑิตพัฒนบริหารศาสตร์ (NIDA)
ปี พ.ศ. 2552 ปริญญาเอก Ph.D. (Computer Science)
University Technology of Malaysia (UTM)

6. สาขาวิชาการที่มีความชำนาญพิเศษ (แตกต่างจากวุฒิการศึกษา) ระบุสาขาวิชาการ
- การจัดการฐานข้อมูล ระบบเครือข่าย
- Grid computing/High performance system.
- Parallel and distributed computing system.
- Simulation modeling.
- Artificial Intelligence Algorithm
- เศรษฐศาสตร์, การท่องเที่ยว, Logistic systems

7. ประสบการณ์ที่เกี่ยวข้องกับการบริหารงานวิจัยทั้งภายในและภายนอกประเทศ โดยระบุสถานภาพในการทำการวิจัยว่าเป็นผู้อำนวยการแผนงานวิจัย หัวหน้าโครงการวิจัย หรือผู้ร่วมวิจัยในแต่ละผลงานวิจัย

7.1 **ผู้อำนวยการแผนงานวิจัย** :

7.2 **หัวหน้าโครงการวิจัย:**

1) ชื่อโครงการวิจัย “ระบบตรวจสอบมาตรฐานสถานประกอบการด้านอาหารแบบออนไลน์ โครงการนี้มี MOU (ระหว่างหลักสูตรวิทยาการคอมพิวเตอร์และสำนักสุขาภิบาลอาหารและน้ำ กรมอนามัย กระทรวงสาธารณสุข)”
2) หัวหน้าโครงการวิจัย “ระบบคลังข้อมูลสถานประกอบการด้านอาหารของประเทศไทย” ของสานักสุขาภิบาลอาหารและนา กรมอนามัย กระทรวงสาธารณสุข (ปัจจุบัน)
3) -หัวหน้าโครงการวิจัย “ระบบสารสนเทศอนามัยสิ่งแวดล้อมของประเทศไทย (NEHIS)” กรมอนามัย กระทรวงสาธาณสุข (ปัจจุบัน)
4) นักวิจัย “การจัดการความรู้ของภูมิปัญญาท้องถิ่นด้านอาหารของชุมชน ตลาดดอนหวาย จ.นครปฐม” สานักงานคณะกรรมการวิจัยแห่งชาติ (ทุน วช.56)

7.3 งานวิจัยที่ทำเสร็จแล้ว: ชื่อผลงานวิจัย ปีที่พิมพ์ การเผยแพร่ และแหล่งทุน (อาจมากกว่า 1 เรื่อง)

1. **Siriluck Lorpunmanee,** Chai Chompoo-inwai, Chow Chompoo-inwai and Witcha Chimphlee. Grid Scheduling System Based on the Dynamic ACO Algorithm. The 33rd Electrical Engineering Conference, 1-3 December 2010.

2. ABOAMAMA ATAHAR AHMED, MUHAMMAD SHAFE ABD LATIFF, KAMALRULNIZAM ABU BAKAR, ABDUL HANAN ABDULLAH, ZAINUL AHMAD RAJION AND **SIRILUCK LORPUNMANEE.** GEVP: GRID ENABLED VISUALIZATION PIPELINE. International Journal of Grid Computing & Applications (IJGCA) Vol.2, No.1, March 2011.

3. **Lorpunmanee Siriluck**, Scheduling in Computational Grid with Hybrid Algorithms to improve Quality of Service (QoS), การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 35 (EECON-35)), นครนายก, 2555

4. มงคล ฤกษ์พงษ์พันธ์, นนทวัฒน์ คงสุวรรณ์, **ศิริลักษณ์ หล่อพันธ์มณี**, **Distributed Visualization Rendering for Medical Datasets on Grid**

Computing Environment, 2st The ASEAN Undergraduate Conference in Computing (AUCC-2557), มหาวิทยาลัยศรีปทุม วิทยาเขตชลบุรี, ไทย.

5. ภทร สว่างศรี, อวยชัย คุณเจริญทรัพย์, **ศิริลักษณ์ หล่อพันธ์มณี**, **ระบบการแจ้งเตือนการล้มของผู้สูงอายุ,** 2st The ASEAN Undergraduate Conference in Computing (AUCC-2557), มหาวิทยาลัยศรีปทุม วิทยาเขตชลบุรี, ไทย.

7.4 งานวิจัยที่กำลังทำ: ชื่อข้อเสนอการวิจัย แหล่งทุน และสถานภาพในการทำวิจัยว่าได้ทำการวิจัยลุล่วงแล้วประมาณร้อยละเท่าใด

ชื่อโครงการ: การจัดการความรู้ของภูมิปัญญาท้องถิ่นด้านอาหารของชุมชนตลาดดอนหวาย จ.นครปฐม, ทุนวช. 2556, รอส่งเล่มสมบูรณ์

ชื่อโครงการ: การทำบาตรของชุมชนบ้านบาตร, ทุนกระทรวงศิลปวัฒนธรรม 2557