

บทที่ 4

การประมวลผลแบบขนานและโครงสร้างของระบบคลัสเตอร์

4.1 บทนำ

แนวความคิดของระบบคลัสเตอร์เกิดขึ้นในปี ค.ศ. 1960 โดยบริษัท IBM ต้องการเชื่อมเมนเฟรม (Mainframe) ขนาดใหญ่เข้าด้วยกันเพื่อใช้ในเชิงการค้า แต่ในตอนนั้นระบบคลัสเตอร์ยังไม่เป็นที่รู้จักอย่างแพร่หลายเนื่องจากข้อจำกัดทางด้านราคาและความแพร่หลายของฮาร์ดแวร์ (Hardware) นอกจากนั้นยังขาดเครื่องมือ (Tools) มาตรฐานสำหรับการเชื่อมต่อเครื่องในระบบเข้าด้วยกัน จนในช่วงกลางปี 1980 เป็นช่วงที่เทคโนโลยีระบบคลัสเตอร์สามารถเกิดขึ้นได้เนื่องจากสามารถสร้างไมโครโปรเซสเซอร์ที่มีประสิทธิภาพได้ และเกิดระบบเครือข่ายความเร็วสูงขึ้น ซึ่งเครื่องคอมพิวเตอร์ในระบบเครือข่ายสามารถสื่อสารกันได้ภายในเวลาไม่เกินหนึ่งมิลลิวินาที นอกจากนั้นความต้องการความเร็วในการประมวลผลงานทางวิทยาศาสตร์และวิศวกรรมศาสตร์ซึ่งมีมากขึ้นอย่างต่อเนื่องก็มีส่วนผลักดันให้เกิดเทคโนโลยีคลัสเตอร์ขึ้น

ระบบคลัสเตอร์ได้เกิดขึ้นอย่างเป็นทางการเป็นรูปธรรมในปี 1994 เมื่อ Thomas Sterling และ Don Becker จาก CESDIS (The Center of Excellence in Space Data and Information Sciences) ต้องการที่จะวิเคราะห์ข้อมูลจากอวกาศที่มีจำนวนมากและซับซ้อนซึ่งต้องใช้เครื่องคอมพิวเตอร์ที่สามารถประมวลผลได้เร็วมาก ๆ และในขณะนั้นมีเพียงซูเปอร์คอมพิวเตอร์ซึ่งมีราคาแพงแต่เนื่องจากงบประมาณที่มีอย่างจำกัดทำให้ Sterling และ Becker มีแนวความคิดที่จะใช้ระบบคลัสเตอร์มาทำการประมวลผล โดยได้สร้างระบบคลัสเตอร์จากเครื่องคอมพิวเตอร์ซึ่งใช้ชิพ Intel DX4 เป็นหน่วยประมวลผลกลาง (CPU) และส่วนประกอบอื่น ๆ ซึ่งสามารถหาได้ง่ายจากท้องตลาดและได้ตั้งชื่อระบบคลัสเตอร์นี้ว่า Beowulf (Sterling and Becker, 1994)

ระบบคลัสเตอร์มักจะเป็นระบบที่แต่ละหน่วยประมวลผลหรือโหนดการคำนวณ (Computing node) อยู่ใกล้กัน หรืออยู่ในพื้นที่ใกล้เคียงกัน แล้วทำการเชื่อมต่อระหว่างโหนดต่อโหนดหรือแต่ละหน่วยประมวลผลผ่านทางเครือข่าย เช่น Gigabit Ethernet, Myrinet, InfiniBand, Quadrics หรือ เครือข่ายรูปแบบอื่น ๆ ซึ่งต้องเป็นเทคโนโลยีเครือข่ายที่มีความเร็วสูง และค่า Latency Time ต่ำ เพื่อให้การติดต่อสื่อสารระหว่างกันสามารถเป็นไปได้อย่างรวดเร็ว และสูญเสียเวลาในเครือข่ายน้อยที่สุดเสมือนว่าทุกหน่วยประมวลผลอยู่ใกล้กันมากจนเกือบจะเหมือนอยู่บนแผงวงจรรวม Mother Board เดียวกัน ทำให้การส่งผ่านข้อความเพื่อแลกเปลี่ยนข้อมูลต่าง ๆ ระหว่างคำนวณเป็นไปอย่างรวดเร็ว

4.2 การประมวลผลแบบขนาน

การประมวลผลแบบขนานเป็นวิธีการในการแบ่งปัญหามาเป็นปัญหาย่อยๆ แล้วทำการแก้ปัญหานั้นไปพร้อมๆ กัน ในปัจจุบันระบบประมวลผลแบบขนานถูกใช้อย่างแพร่หลายทั้งในการคำนวณทางวิทยาศาสตร์และการประยุกต์ใช้กับงานทั่วไป จากความต้องการความสามารถในการประมวลผลที่มากขึ้นนี้ทำให้มีความต้องการระบบคอมพิวเตอร์ที่มีความเร็วสูงขึ้น ราคาถูกลง และมีการพัฒนาอย่างต่อเนื่อง และในการพัฒนาระบบคอมพิวเตอร์แบบขนานถูกแบ่งออกเป็นสองกลุ่มคือ

1. ระบบคอมพิวเตอร์ขนานแบบหลายหน่วยประมวลผล (Massively Parallel Processors (MPPs)) ซึ่งเป็นระบบที่มีความสามารถสูงที่สุดในปัจจุบันโดยระบบคอมพิวเตอร์นี้อาจประกอบด้วยหน่วยประมวลผลกลางตั้งแต่หนึ่งร้อยถึงหนึ่งพันหน่วยซึ่งหน่วยประมวลผลกลางเชื่อมต่อกันผ่านระบบบัส (BUS) ปัจจุบันระบบ MPP ถูกใช้ในการแก้ปัญหที่ต้องการความเร็วในการประมวลผลมากๆ หลายอย่าง เช่น การสร้างแบบจำลองสภาพภูมิอากาศของโลก และ การออกแบบยาเป็นต้น

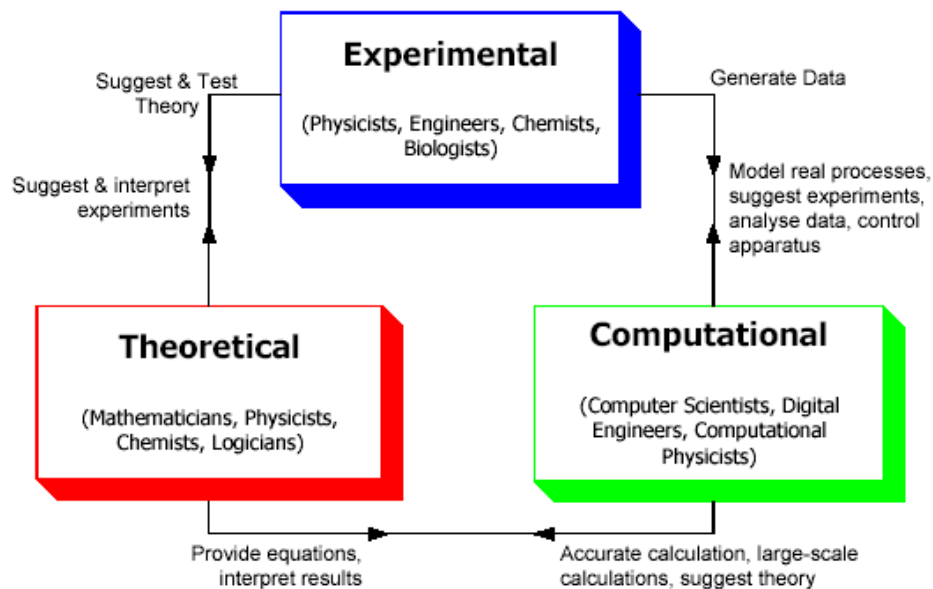
2. ระบบคอมพิวเตอร์แบบกระจาย (Distributed Computing) ซึ่งเป็นกระบวนการที่เครื่องคอมพิวเตอร์หลายๆ เครื่องเชื่อมต่อกันผ่านระบบเครือข่ายเพื่อแก้ปัญหามาเป็นปัญหาย่อยๆ โดยประสิทธิภาพของระบบคอมพิวเตอร์แบบนี้อาจจะเทียบเท่ากับหรือมากกว่าระบบซูเปอร์คอมพิวเตอร์ขนาดใหญ่ได้ ระบบระบบคอมพิวเตอร์ขนานแบบหลายหน่วยประมวลผลที่ได้กล่าวถึงในตอนแรกสามารถนำมาเชื่อมต่อกันผ่านทางระบบเครือข่ายเพื่อใช้ในลักษณะของระบบระบบคอมพิวเตอร์แบบกระจายเพื่อทำให้มีประสิทธิภาพในการคำนวณมากขึ้น

4.2.1 การประยุกต์ใช้การประมวลผลแบบขนาน

ในอดีตการแก้ปัญหามาเป็นปัญหาย่อยๆ ทางวิทยาศาสตร์มีสาขาวิชาที่สัมพันธ์กันสองสาขา คือ วิทยาศาสตร์เชิงทฤษฎี (Theoretical Science) และ วิทยาศาสตร์เชิงการทดลอง (Experimental Science) นักวิทยาศาสตร์เชิงทฤษฎีจะใช้ทฤษฎีทางวิทยาศาสตร์สร้างแบบจำลองทางคณิตศาสตร์และสมมติฐานของปัญหา จากนั้น นักวิทยาศาสตร์เชิงการทดลองนำแบบจำลองและสมมติฐานของปัญหาไปสร้างการทดลองเพื่อพิสูจน์แบบจำลองและสมมติฐาน ถ้าหากการทดลองเป็นไปตามสมมติฐานและแบบจำลองถูกต้องแสดงว่าสามารถแก้ปัญหานั้นๆ ได้ แต่ถ้าหากการทดลองล้มเหลว นักวิทยาศาสตร์เชิงทฤษฎีจะนำแบบจำลองและสมมติฐานกลับไปแก้ไขและนำกลับมาทดลองอีกครั้ง ปัญหาในปัจจุบันมีความซับซ้อนมาก งานหลายอย่างไม่สามารถเสี่ยงต่อการล้มเหลวได้ เนื่องจากข้อจำกัดทางด้านงบประมาณ เวลา และความปลอดภัย เมื่อระบบคอมพิวเตอร์ถูกพัฒนาขึ้นมาจึงเกิดสาขาวิชาวิทยาศาสตร์เชิงคำนวณ (Computational Science) ขึ้นมาเพื่อช่วยในการแก้ปัญหามาเป็นปัญหาย่อยๆ ทางวิทยาศาสตร์ที่ซับซ้อนให้เป็นไปอย่างมีประสิทธิภาพ ความสัมพันธ์ของวิทยาศาสตร์ทั้งสาม

สาขานี้แสดงดังรูปที่ 4.1 เมื่อนักวิทยาศาสตร์เชิงทฤษฎีพัฒนาแบบจำลองทางคณิตศาสตร์ของปัญหาขึ้นมาจะส่งแบบจำลองนั้นให้แก่นักวิทยาศาสตร์เชิงคำนวณเพื่อสร้างโปรแกรมเพื่อแก้ปัญหานั้นแบบเชิงตัวเลข (Numerically) ส่วนนักวิทยาศาสตร์เชิงการทดลองจะให้คำแนะนำเกี่ยวกับคุณสมบัติในความเป็นจริงต่างๆที่ใช้ในการแก้ปัญหานั้นแก่นักวิทยาศาสตร์เชิงคำนวณเพื่อใช้เป็นข้อมูลในการแก้ปัญหานั้น ข้อดีของการใช้คอมพิวเตอร์แก้ปัญหานั้นทางวิทยาศาสตร์มีดังนี้

1. ใช้งบประมาณน้อยกว่า และทำได้รวดเร็วกว่าการทดลองในห้องปฏิบัติการ
2. คอมพิวเตอร์สามารถแก้ปัญหานั้นได้หลายเงื่อนไขกว่าการใช้เฉพาะเครื่องมือที่มีอยู่ในห้องปฏิบัติการ
3. ข้อจำกัดของการแก้ปัญหานั้นด้วยการคำนวณมีเพียงแค่ความเร็วและจำนวนหน่วยความจำของคอมพิวเตอร์ แต่การทดลองในห้องปฏิบัติการจะมีข้อจำกัดในทางปฏิบัติมากกว่า



รูปที่ 4.1 ความสัมพันธ์ระหว่างวิทยาศาสตร์เชิงทฤษฎี การทดลอง และเชิงคำนวณ

การประยุกต์ใช้ระบบการประมวลผลแบบขนานสามารถแบ่งออกได้เป็นห้ากลุ่มใหญ่ๆคือ

4.2.1.1 แบบจำลองของการทำนายและการจำลองแบบ (Predictive Modelling and Simulations)

แบบจำลองหลายมิติ (Multidimensional Modeling) ของอากาศ สิ่งแวดล้อมของโลก อวกาศและเศรษฐกิจโลก การสร้างแบบจำลองเพื่อทำนายเหตุการณ์เหล่านี้ถูกทำโดยการจำลองด้วยคอมพิวเตอร์ซึ่งการคำนวณต้องการระบบที่มีความเร็วในการประมวลผลอย่างต่ำ 1 GFLOPS (Giga-floating point operation per second) หรือหนึ่งพันล้านการปฏิบัติงานแบบทศนิยมต่อหนึ่งวินาที

4.2.1.2 การออกแบบทางวิศวกรรมและระบบอัตโนมัติ (Engineering Design and Automation)

ระบบคอมพิวเตอร์แบบขนานถูกใช้ในการวิเคราะห์และออกแบบทางวิศวกรรมเช่น การวิเคราะห์ไฟไนต์อีลีเมนต์ (Finite-Element Analysis) พลศาสตร์เชิงคำนวณ (Computational Aerodynamic) เป็นต้น

4.2.1.3 การค้นหาแหล่งพลังงาน (Energy Resources Exploration)

คอมพิวเตอร์มีบทบาทสำคัญในการค้นหาน้ำมันและก๊าซธรรมชาติทั้งในด้านการขุดหา และการจัดการควบคุมระบบ

4.2.1.4 ด้านการแพทย์ การทหาร และการวิจัยพื้นฐาน (Medical, Military and Basic Research)

ในด้านการแพทย์ คอมพิวเตอร์ความเร็วสูงจะถูกใช้ในการสร้างภาพทางการแพทย์ การออกแบบหัวใจเทียม การวินิจฉัยโรคตับ และการศึกษาทางด้านพันธุวิศวกรรม สำหรับด้านการทหารคอมพิวเตอร์ความเร็วสูงจะถูกใช้ในการออกแบบอาวุธ และการทำสงครามด้วยอุปกรณ์อิเล็กทรอนิกส์

4.2.1.5 การสร้างภาพ (Visualization)

เทคนิคซึ่งใช้ในอุตสาหกรรมหนังเช่น The Matrix และ Toy Story ซึ่งการคำนวณสำหรับสร้างภาพต้องใช้คอมพิวเตอร์ความเร็วสูงสำหรับสร้างภาพเสมือนจริงจำนวนมาก ในหนังทั้งสองเรื่องใช้ระบบคลัสเตอร์ในการประมวลผล นอกจากนั้นการสร้างภาพเสมือนจริงจากข้อมูลทางวิทยาศาสตร์ก็เป็นอีกงานหนึ่งซึ่งต้องใช้คอมพิวเตอร์ความเร็วสูงในการประมวลผลเพื่อสร้างภาพ

4.3 การจำแนกประเภทของคอมพิวเตอร์

โดยทั่วไปแล้วสามารถจำแนกชนิดของระบบคลัสเตอร์ออกตามกฎเกณฑ์ต่างๆดังนี้[10]

4.3.1 จำแนกตามงานที่ประยุกต์ใช้ (Application Target)

ระบบคลัสเตอร์สามารถนำมาไปประยุกต์ใช้งานได้หลายอย่าง ไม่ว่าจะเป็นงานทางด้านการคำนวณหรืองานทางด้านการเป็นเครื่องแม่ข่ายให้บริการงานต่างๆ โดยสามารถแบ่งย่อยได้เป็นสองประเภทคือ

- ระบบคลัสเตอร์แบบประสิทธิภาพสูง (High Performance (HP) Clusters) ระบบนี้มักจะนำมาประยุกต์ใช้ในการคำนวณทางด้านวิทยาศาสตร์และคณิตศาสตร์ ระบบคลัสเตอร์แบบนี้จะถูกสร้างขึ้นมาให้มีความรวดเร็วในการคำนวณมากที่สุด ประสิทธิภาพของหน่วยประมวลผล

จะต้องสูงเพียงพอ อีกทั้งเครือข่ายที่ใช้ในการเชื่อมต่อต้องมีคุณภาพดีมาก ประสิทธิภาพในการคำนวณจึงจะสูงตามไปด้วย

- ระบบคลัสเตอร์แบบเสถียรภาพสูง (High Availability (HA) Clusters) ระบบคลัสเตอร์แบบนี้จะเน้นไปทางด้านเครื่องแม่ข่ายที่ให้บริการงานต่าง ๆ เช่น ให้บริการเป็นเว็บเซิร์ฟเวอร์ (Web Server) หรือให้บริการพื้นที่เก็บข้อมูลบนเครือข่าย (Storage Server) เพื่อจะทำให้มั่นใจได้ว่าผู้ใช้งานทั่ว ๆ ไป จะสามารถเข้าถึงทรัพยากรต่าง ๆ ได้ตลอดเวลา การให้บริการไม่ได้ขึ้นอยู่กับเครื่องเซิร์ฟเวอร์เครื่องใดเครื่องหนึ่ง

4.3.2 จำแนกตามลักษณะของเครื่องในระบบ

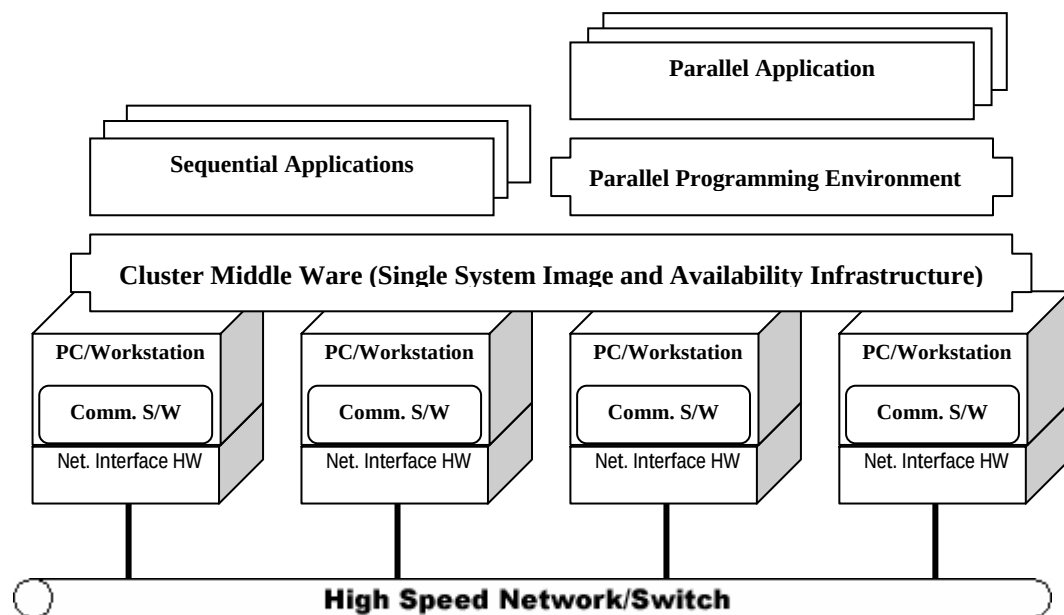
ระบบคลัสเตอร์เกิดจากการเชื่อมต่อเครื่องคอมพิวเตอร์หลายๆเครื่องเข้าด้วยกันโดยเครื่องคอมพิวเตอร์เหล่านี้อาจจะมีลักษณะเหมือนกันทั้งหมดหรือไม่เหมือนกันเลยก็ได้ การจำแนกนี้สามารถแบ่งระบบคลัสเตอร์ออกเป็นสองพวกคือ

- ระบบคลัสเตอร์แบบเนื้อเดียว (Homogeneous Cluster) ระบบคลัสเตอร์แบบนี้เป็นระบบที่มีองค์ประกอบทางด้านฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในแต่ละเครื่องเหมือนกันทั้งหมดได้แก่หน่วยประมวลผลกลาง ชนิดและขนาดของหน่วยความจำ ชนิดและขนาดของฮาร์ดดิสก์ และชนิดของระบบปฏิบัติการเป็นต้น โดยระบบคลัสเตอร์นี้เป็นแบบที่นิยมสร้างกันเนื่องจากการบริหารจัดการระบบสามารถทำได้อย่างสะดวก นอกจากนั้นการเขียนโปรแกรมเพื่อทำการประมวลผลบนระบบคลัสเตอร์สามารถเขียนในครั้งเดียวแล้วทำงานได้กับทุกๆเครื่องในระบบ

- ระบบคลัสเตอร์แบบเนื้อผสม (Heterogeneous Cluster) ระบบคลัสเตอร์แบบนี้เป็นระบบที่มีความยืดหยุ่นสูงโดยสามารถสร้างจากเครื่องแบบใดก็ได้ที่สนับสนุนการประมวลผลแบบขนาน แต่ปัญหาของระบบนี้คือการสร้างโปรแกรมสำหรับประมวลผลแบบขนานจะมีความยุ่งยากและซับซ้อนมากขึ้นเนื่องจากต้องทำการสร้างโปรแกรมที่สามารถประมวลผลเฉพาะกับเครื่องหนึ่งๆ เช่นเครื่องที่ใช้ระบบปฏิบัติการลินุกซ์ (Linux) และระบบที่ใช้ไมโครซอฟท์วินโดวส์ (Microsoft Windows) เป็นระบบปฏิบัติการ โปรแกรมที่สร้างขึ้นมาอาจไม่สามารถใช้ด้วยกันได้ วิธีแก้ปัญหอย่างหนึ่งคือสร้างโปรแกรมโดยใช้ตัวแปรภาษาที่สนับสนุนมาตรฐานเช่น ANSI C เป็นต้น หรืออาจสร้างโปรแกรมที่สามารถบนประมวลผลได้บนทุกระบบโดยที่ใช้รหัสต้นฉบับ (Source Code) ตัวเดียวกัน เช่นภาษาจาวา (Java) เป็นต้น

4.4 องค์ประกอบและการสร้างระบบคลัสเตอร์

องค์ประกอบของระบบคลัสเตอร์แสดงดังรูปที่ 4.2



รูปที่ 4.2 องค์ประกอบของระบบคลัสเตอร์

รายละเอียดของแต่ละองค์ประกอบมีดังนี้

4.4.1 ระบบเครือข่ายความเร็วสูง (High Speed Network)

การสื่อสารระหว่างแต่ละเครื่องในระบบคลัสเตอร์จะผ่านระบบเครือข่ายความเร็วสูง ซึ่งอุปกรณ์เครือข่ายแต่ละชนิดจะมีความเร็วและราคาแตกต่างกันไป ตัวอย่างอุปกรณ์เครือข่ายที่สามารถใช้สร้างระบบคลัสเตอร์ได้มีดังนี้

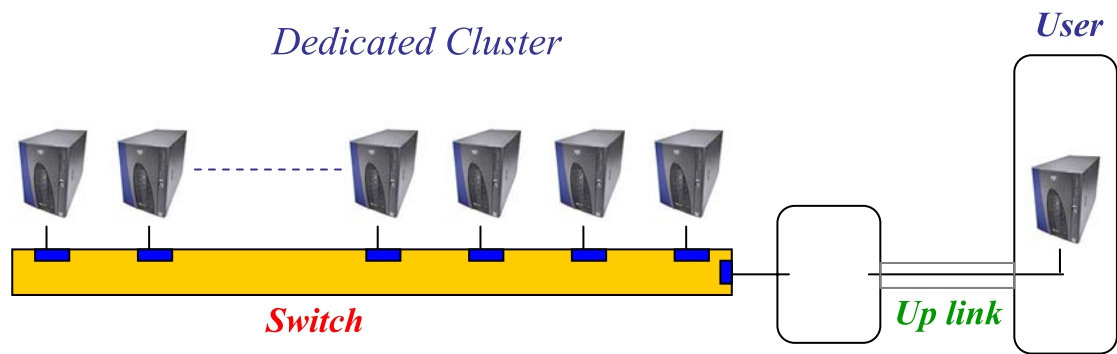
- Ethernet ในปัจจุบันอุปกรณ์ Ethernet ได้ถูกพัฒนาให้มีความเร็วในการรับส่งข้อมูลสูงมากขึ้นจนถึงระดับ Gigabit Ethernet หรือ 10 Gigabit Ethernet คือ มีความเร็วในการส่งผ่านข้อมูลประมาณ 1-10 พันล้านบิตต่อวินาที

- Myrinet มีความเร็วในการส่งผ่านข้อมูล 2 พันล้านบิตต่อวินาที (Gigabit per second) และมีค่า Latency Time ต่ำกว่าเครือข่ายแบบ Ethernet มาก แต่ก็มีราคาแพงกว่า Ethernet มาก ในระบบคลัสเตอร์ที่ต้องการความเร็วในการส่งข้อมูลสูงมากๆ มักใช้ระบบเครือข่ายนี้ในการสื่อสารระหว่างเครื่อง

- Quadrics มีความเร็วในการส่งข้อมูลอยู่ที่ 340-900 MB/second หรือประมาณ 2.65-7 Gbit/sec และค่า Latency time มีค่าต่ำมาก

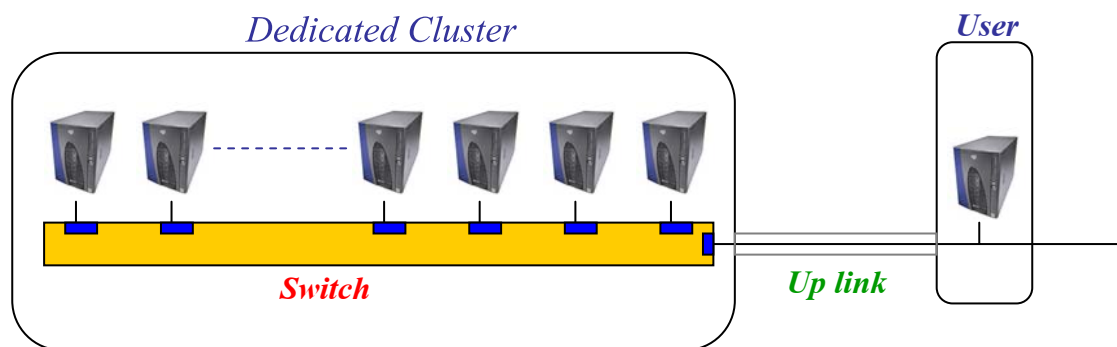
- InfinBand เป็นเทคโนโลยีที่มีความเร็วในการสื่อสารข้อมูลสูงมากถึง 5 Gbit/sec สำหรับคาร์ดรุ่นที่มีความเร็ว 4x อีกทั้งค่า Latency Time มีค่าต่ำมาก น้อยกว่า 10 microsecond

การเชื่อมต่อแต่ละเครื่องเข้ากับระบบเครือข่ายอาจทำได้ดังรูปที่ 4.3 โดยมีเครื่องเดียวที่เชื่อมต่อกับระบบภายนอกซึ่งอาจเป็นระบบอินเทอร์เน็ตก็ได้ ส่วนเครื่องที่เหลือต่อเข้ากับ Hub หรือ Switching ธรรมดา ลักษณะการเชื่อมต่อแบบรูปที่ 4.4 มีข้อดีคือมีความปลอดภัยสูง ผู้ใช้จะต้องผ่านเครื่องที่ต่อกับเครือข่ายภายนอกก่อนถึงจะสามารถเข้าถึงเครื่องที่อยู่ในเครือข่ายภายในได้



รูปที่ 4.3 การเชื่อมต่อเครื่องในระบบคลัสเตอร์เข้ากับระบบเครือข่ายส่วนตัว (Private Network)

ถึงแม้ว่าระบบในรูปที่ 4.3 จะมีความปลอดภัยสูงแต่ในแง่การประยุกต์ใช้กับงานบางอย่างเช่นงานที่ต้องการเข้าถึงแต่ละเครื่องโดยตรงอาจทำได้ไม่สะดวกเช่นงานเกี่ยวกับศูนย์รวมเซิร์ฟเวอร์ (Server Farm) ในการทำอีเมลล์เซิร์ฟเวอร์ที่มีความคงทนสูง (High Availability Email Server) หรือ เว็บเซิร์ฟเวอร์ที่มีความคงทนสูง (High Availability Web Server) สามารถทำการเชื่อมต่อแต่ละเครื่องเข้ากับระบบเครือข่ายได้ดังรูปที่ 4.4 ซึ่งแต่ละเครื่องเชื่อมต่อกันผ่านระบบเครือข่ายในขณะที่ทุกๆเครื่องก็เชื่อมต่อกับระบบภายนอกด้วยเช่นกัน



รูปที่ 4.4 การเชื่อมต่อระบบเครือข่ายแบบศูนย์รวมเซิร์ฟเวอร์

4.4.2 โปรแกรมบริการการสื่อสาร (Communication Software)

เนื่องจากในวิทยานิพนธ์ฉบับนี้ใช้ MPI เป็นโปรแกรมสำหรับสร้างสภาวะแวดล้อมของการโปรแกรมแบบขนาน (Parallel Programming Environment) ซึ่งจะกล่าวถึงในหัวข้อที่ 4.4.5 โดย MPI สนับสนุนโปรแกรมบริการการสื่อสารสองตัวคือ RSH (Remote Shell) และ SSH (Secure Shell) โปรแกรมทั้งสองตัวนี้ทำหน้าที่จัดการไฟล์และประมวลผลโปรแกรมระหว่างเครื่องได้ Remote Shell จะถูกติดตั้งมาให้ในระบบปฏิบัติการแบบยูนิกซ์ (UNIX) ส่วนมาก แต่มีข้อบกพร่องในเรื่องระบบรักษาความปลอดภัย ในระบบปฏิบัติการแบบยูนิกซ์ยุคใหม่มักไม่แนะนำให้ใช้ Remote Shell เช่นในระบบปฏิบัติการลินุกซ์ (Linux) ส่วนมากจะมี Secure Shell ติดตั้งมาให้ซึ่งมีความปลอดภัยมากกว่า Remote Shell เนื่องจากการเข้ารหัสข้อมูลก่อนทำการส่งข้อมูล ถ้าระบบคลัสเตอร์ไม่ได้เชื่อมต่อกับระบบอินเทอร์เน็ตก็เป็นไปได้ว่า Remote Shell สามารถใช้ได้อย่างปลอดภัย

4.4.3 ระบบปฏิบัติการ (Operating System)

ระบบปฏิบัติการที่สามารถใช้ระบบเครือข่ายได้ส่วนมากสามารถนำมาสร้างเป็นระบบคลัสเตอร์ได้ เช่น ไมโครซอฟท์วินโดวส์ เอ็นที (Microsoft Windows NT) ไมโครซอฟท์วินโดวส์สองพัน (Microsoft Windows 2000) ลินุกซ์ (Linux) ซันโซลาริส (Sun Solaris) เป็นต้น ระบบปฏิบัติการอื่นๆที่เป็นระบบยูนิกซ์สามารถใช้ร่วมกับระบบลินุกซ์เพื่อทำเป็นระบบคลัสเตอร์ได้ง่าย เนื่องจากมีหลักการออกแบบและโครงสร้างของระบบปฏิบัติการใกล้เคียงกัน

4.4.4 คลัสเตอร์มิดเดิลแวร์ (Cluster Middle Ware)

คลัสเตอร์มิดเดิลแวร์เป็นโปรแกรมที่ทำให้ผู้ใช้ระบบคลัสเตอร์รู้สึกว่าการใช้งานระบบเดี่ยวเช่นเดียวกับที่ใช้เครื่องคอมพิวเตอร์ส่วนบุคคลอยู่ โปรแกรมส่วนคลัสเตอร์มิดเดิลแวร์นี้จะต้องทำที่ระดับแก่น (Kernel) ของระบบปฏิบัติการเพื่อที่จะสามารถจัดการถ่ายโอนโปรเซส (Process) ที่เกิดขึ้นไปสู่เครื่องอื่นๆในระบบได้ ตัวอย่างของคลัสเตอร์มิดเดิลแวร์นี้คือ

- BPROC (Beowulf-distributed process space) ซึ่งเป็นโปรแกรมส่วนที่เพิ่มเข้าไปในแก่นของระบบปฏิบัติการลินุกซ์ โดยพัฒนาจากผู้สร้างระบบ Beowulf ในโครงการ CESDIS ของ NASA
- MOSIX เป็นโปรแกรมส่วนที่เพิ่มเข้าไปในส่วน of แก่นของระบบปฏิบัติการเช่นเดียวกับ BPROC ถูกพัฒนาโดยมหาวิทยาลัย Hebrew University of Jerusalem
- Solaris-MC เป็นระบบปฏิบัติการที่สนับสนุนการกระจายงานในระบบคลัสเตอร์ถูกพัฒนาโดย Sun Labs

เนื่องจากว่าระบบคลัสเตอร์ที่ใช้ในวิทยานิพนธ์ฉบับนี้เป็นระบบคลัสเตอร์ที่สร้างขึ้นเฉพาะงาน การกระจายงานอาจไม่จำเป็นต้องใช้คลัสเตอร์มิดเดิลแวร์ แต่การกระจายงานจะทำในชั้นของสถานะแวดล้อมของการโปรแกรมแบบขนานเช่น MPI หรือ PVM

4.4.5 โปรแกรมสำหรับสร้างสถานะแวดล้อมของการโปรแกรมแบบขนาน (Parallel Programming Environment)

โปรแกรมสำหรับสร้างสถานะแวดล้อมของการโปรแกรมแบบขนานมีหน้าที่ทำให้โปรแกรมแบบขนานที่ประมวลผลผ่านสถานะแวดล้อมนี้สามารถกระจายงานได้อย่างอัตโนมัติตามที่ผู้สร้างโปรแกรมได้ออกแบบไว้ สถานะแวดล้อมของการโปรแกรมแบบขนานแบ่งออกเป็นสองส่วนคือ

4.4.5.1 ส่วนของโปรแกรมสร้างสถานะแวดล้อมแบบขนาน

ส่วนของโปรแกรมสถานะแวดล้อมนี้ทำหน้าที่ในการเชื่อมต่อระหว่างเครื่องในระบบคลัสเตอร์ เมื่อทำการประมวลผลโปรแกรมแบบขนานที่ออกแบบโดยใช้คลังชุดคำสั่ง (Library) ของการโปรแกรมแบบขนานนี้ สถานะแวดล้อมแบบขนานจะทำการส่งงานที่ผู้สร้างโปรแกรมกำหนด ไปประมวลผลบนเครื่องที่เลือกไว้โดยอัตโนมัติ หรือถ้าในโปรแกรมที่สร้างขึ้นไม่ได้ระบุชื่อของเครื่องที่ต้องการให้ประมวลผล สถานะแวดล้อมแบบขนานนี้จะทำการกระจายงานให้โดยอัตโนมัติคล้ายกับหน้าที่ของคลัสเตอร์มิดเดิลแวร์ แต่ส่วนของสถานะแวดล้อมแบบขนานนี้มีความยืดหยุ่นและประสิทธิภาพมากกว่า แต่ก็สร้างได้ยากกว่าเนื่องจากขึ้นอยู่กับโปรแกรมประยุกต์ที่สร้างขึ้น

4.4.5.2 ส่วนคลังชุดคำสั่ง (Library) ของการโปรแกรมแบบขนาน

โปรแกรมแบบขนานจำเป็นต้องทำการสื่อสารผ่านระบบเครือข่าย ทางเลือกหนึ่งที่สะดวกรวดเร็วและมีประสิทธิภาพคือใช้คลังชุดคำสั่งแบบขนานที่มีผู้ที่พัฒนาไว้แล้ว การกระจายงานอาจประกอบด้วยการกระจายให้เครื่องทุกเครื่องในระบบหรือกระจายสู่บางเครื่อง หรือกระจายสู่กลุ่มของเครื่องที่มีระบบปฏิบัติการเดียวกันหรือกลุ่มของเครื่องที่ได้ออกแบบไว้ก่อน นอกจากนั้นคลังชุดคำสั่งยังได้จัดเตรียมวิธีการส่งข้อมูลชนิดต่างๆเช่น จำนวนเต็ม ทศนิยม หรือแม้แต่ข้อมูลที่มีความสลับซับซ้อนมากๆเช่นข้อมูลในหน่วยความจำที่ไม่อยู่ติดเป็นกลุ่มเดียวเป็นต้น

โปรแกรมในส่วนของสถานะแวดล้อมของการโปรแกรมแบบขนานที่นิยมใช้ในงานวิจัยมีสองตัวคือ

- Parallel Virtual Machine (PVM)ถูกพัฒนาโดย Oak Ridge National Laboratory มีลักษณะเป็นเครื่องเสมือน (Virtual Machine) เพื่อใช้สำหรับประมวลผลโปรแกรมแบบขนานเริ่มแรกผู้ใช้ต้องทำการเพิ่มเครื่องในระบบคลัสเตอร์เข้าไปในส่วนที่เป็นเครื่องเสมือนจากนั้นเมื่อทำ

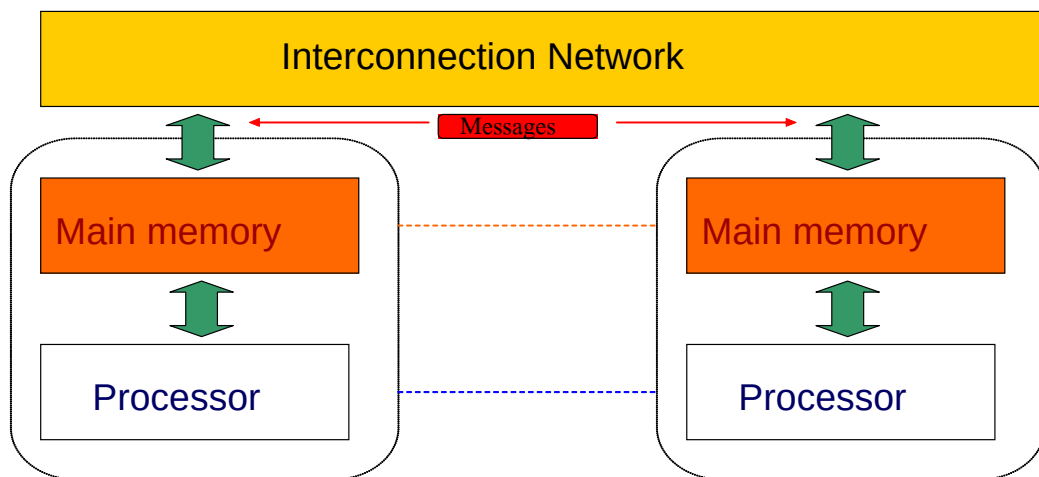
การประมวลผลโปรแกรมส่วนที่เป็นเครื่องเสมือนจะทำการกระจายงานไปทำบนเครื่องที่ต้องการตามที่ได้ออกแบบไว้ในโปรแกรม

- Message Passing Interface (MPI) เป็นมาตรฐานของการประมวลผลแบบขนานในระบบคลัสเตอร์ ซึ่งสนับสนุนภาษา C และ Fortran ในวิทยานิพนธ์ฉบับนี้ใช้ MPICH ซึ่งเป็นคลังชุดคำสั่งของการส่งข้อมูลที่สร้างตามมาตรฐาน MPI ลักษณะของการใช้คลังชุดคำสั่งและการโปรแกรมแบบขนานในลักษณะต่างๆมีดังนี้

4.5 การสื่อสารระหว่างงานด้วยวิธีส่งข้อความ (Message Passing)

จากบทที่ 4 ได้กล่าวถึงโครงสร้างของคอมพิวเตอร์แบบขนานซึ่งระบบคลัสเตอร์เป็นระบบคอมพิวเตอร์แบบขนานที่สื่อสารระหว่างงานด้วยการส่งข้อความ และแต่ละเครื่องมีหน่วยความจำเป็นของตัวเอง (DMMP) ซึ่งแสดงดังรูปที่ 4.5 วิธีการส่งข้อความในระบบแบบหลายคอมพิวเตอร์ (Multicomputer) สามารถทำได้โดย

- ออกแบบตัวแปลภาษาแบบพิเศษซึ่งสนับสนุนการประมวลผลแบบขนาน
- เพิ่มไวยากรณ์ (Syntax) ในตัวแปลภาษาที่มีอยู่แล้วให้สนับสนุนการสื่อสารแบบส่งข้อความ
- ใช้ตัวแปลภาษาที่มีอยู่แล้วร่วมกับคลังชุดคำสั่ง (Library) ซึ่งสนับสนุนการสื่อสารแบบส่งข้อความ



รูปที่ 4.5 แต่ละเครื่องในระบบคลัสเตอร์มีหน่วยความจำของตัวเอง

ตัวแปลภาษาที่สนับสนุนการสื่อสารแบบส่งข้อความได้แก่ OCCAM ซึ่งออกแบบมาเพื่อใช้กับหน่วยประมวลผลที่เรียกว่า Transputer ส่วนตัวแปลภาษาที่มีการเพิ่มไวยากรณ์ของการ

สื่อสารด้วยการส่งข้อความได้แก่ C C++ และ Fortran M ในวิทยานิพนธ์ฉบับนี้ใช้คลัสเตอร์คำสั่งของการสื่อสารแบบส่งข้อความร่วมกับตัวแปลภาษาที่มีอยู่แล้วโดยคลัสเตอร์คำสั่งนี้มีชื่อว่า MPI (Message Passing Interface) คลัสเตอร์คำสั่งอีกตัวหนึ่งที่นิยมใช้กันคือ PVM (Parallel Virtual Machine)

ก่อนที่ระบบคอมพิวเตอร์จะทำการประมวลผลงานจะต้องมีการสร้างโปรเซสขึ้น โดยโปรเซสจะเกิดจากระบบปฏิบัติการ การสร้างโปรเซสในระบบคลัสเตอร์จะถูกทำโดยระบบปฏิบัติการของแต่ละหน่วยประมวลผลเช่นเดียวกันกับระบบคอมพิวเตอร์อื่นๆ การควบคุมการสร้างโปรเซสจะทำจากเครื่องที่ถูกกำหนดเป็นผู้จัดการ(Manager) หรือเครื่องที่ทำการส่งประมวลผลโปรแกรม โดยทั่วไปการสร้างงานในระบบคลัสเตอร์ที่ใช้คลัสเตอร์คำสั่งในการสื่อสารระหว่างงานจะมีสองแบบคือ

- การสร้างโปรเซสแบบกำหนดไว้ก่อน (Static Process Creation) วิธีการสร้างโปรเซสแบบนี้จะใช้ในคลัสเตอร์คำสั่ง MPI ซึ่งการสร้างโปรเซสจะเกิดตอนเริ่มสั่งให้ประมวลผลโปรแกรม

- การสร้างโปรเซสแบบกำหนดในขณะโปรแกรมทำงาน (Dynamic Process Creation) ซึ่งการสร้างโปรเซสในลักษณะนี้จะเกิดเมื่อใดก็ได้เมื่อโปรเซสที่เป็นผู้จัดการยังทำงานอยู่ การสร้างโปรเซสลักษณะนี้จะใช้ในคลัสเตอร์คำสั่ง PVM

ใน MPI จะใช้แบบจำลองโปรแกรมแบบโปรแกรมเดียวหลายข้อมูล (Single Program Multiple Data) หรือ SPMD ซึ่งทุกๆ โปรเซสที่ถูกสร้างในระบบคลัสเตอร์จะประมวลผลโปรแกรมเดียวกันแต่จะสามารถควบคุมให้โปรเซสที่อยู่ในหน่วยประมวลผลที่ต่างกันสามารถประมวลผลในส่วน of โปรแกรมที่ต่างกันได้ ดังแสดงในรูปที่ 4.6

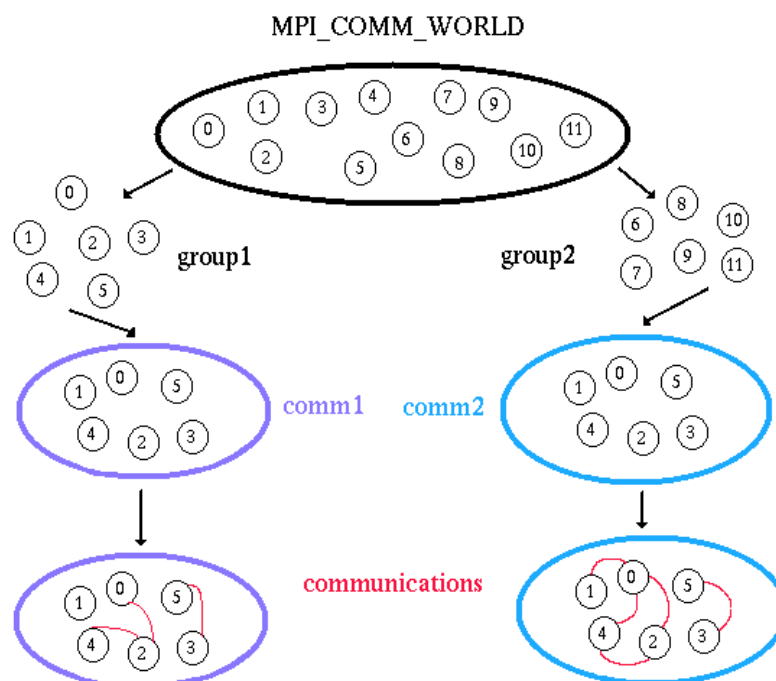
```
int main(int argc, char *argv[])
{
    if(tid == 0){
        ... Manager Code ...
    }else{
        ... Worker Code ...
    }
}
```

รูปที่ 4.6 ลักษณะรหัสต้นฉบับซึ่งมีโครงสร้างแบบโปรแกรมเดียวหลายข้อมูล

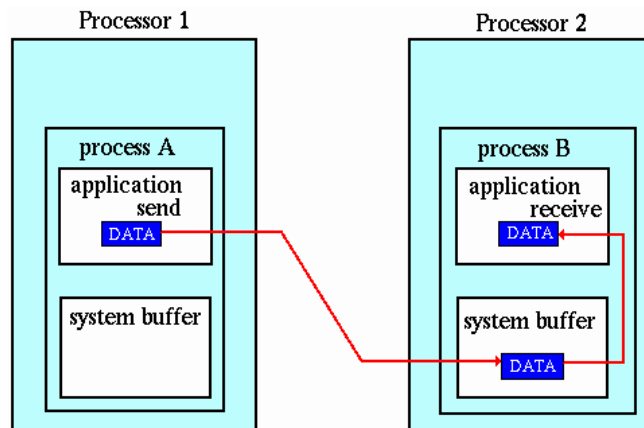
จากรูปที่ 4.6 เป็นรหัสต้นฉบับของโปรแกรมแบบขนาน ค่า tid ที่ได้จากคำสั่ง MPI จะบอกว่าโปรแกรมกำลังประมวลผลอยู่บนโปรเซสใด ถ้า tid มีค่าเท่ากับศูนย์แสดงว่าโปรแกรมกำลังถูกประมวลผลอยู่บนโปรเซสที่ทำหน้าที่เป็นผู้จัดการ ส่วนโปรเซสที่มีค่า tid อื่นๆจะเป็นโปรเซสที่มีหน้าที่เป็นพนักงาน จากคุณสมบัติของค่า tid ดังกล่าวนี้นี้จะทำให้สามารถใส่รหัสต้นฉบับของโปรแกรมแบบขนานสำหรับงานต่างๆในระบบคลัสเตอร์ไว้ในโปรแกรมเดียวได้โดยใช้ประโยคเงื่อนไข if else เป็นตัวแบ่งแยก

งานใน MPI สามารถจัดรวมกลุ่มได้ด้วยสิ่งที่เรียกว่า COMMUNICATOR งานทั้งหมดจะถูกระบุด้วยตัวแปร MPI_COMM_WORLD ผู้สร้างโปรแกรมสามารถกำหนดกลุ่มของงานเพิ่มเติมเพื่อให้สามารถจัดการงานเหล่านั้นได้ง่ายขึ้นดังแสดงในรูปที่ 4.7

จากหัวข้อที่ 5.3.2 ซึ่งเป็นการออกแบบลักษณะการสื่อสารระหว่างงานในระบบคลัสเตอร์ MPI ได้จัดเตรียมคำสั่งสำหรับการส่งและรับข้อมูลลักษณะต่างๆ ซึ่งสามารถประยุกต์ใช้ในโปรแกรมแบบขนานที่ออกแบบไว้ ลักษณะของการส่งและรับข้อมูลของ MPI เป็นดังรูปที่ 4.8 ซึ่งเป็นการส่งข้อมูลจากโปรเซส A ของหน่วยประมวลผล 1 ไปสู่โปรเซส B ในหน่วยประมวลผล 2 การส่งข้อมูลจะส่งโดยตรงจากตัวแปรของโปรแกรมส่วนการรับข้อมูลจะทำผ่านตัวเก็บข้อมูลของระบบ (Buffer) ก่อนแล้วจึงคัดลอกสู่ตัวแปรของโปรแกรมอีกครั้งหนึ่งเพื่อให้สามารถสื่อสารแบบไม่ประสานกันได้ (Asynchronous) การสื่อสารของ MPI สามารถแบ่งได้เป็นการสื่อสารแบบจุดต่อจุด (Point-Point Communication) และแบบสื่อสารเป็นกลุ่ม (Collective Communication)



รูปที่ 4.7 การจัดกลุ่มงานใน MPI



Path of a message buffered at the receiving process

รูปที่ 4.8 การส่งและรับข้อมูลของ MPI

4.5.1 การสื่อสารแบบจุดต่อจุด (Point-Point Communication)

การสื่อสารแบบจุดต่อจุดนี้เป็นพื้นฐานของระบบสื่อสารแบบส่งข้อความ ซึ่งข้อความจะถูกส่งระหว่างโปรเซสสองโปรเซสโดยด้านหนึ่งเป็นผู้ส่งและอีกด้านหนึ่งเป็นผู้รับ คลังชุดคำสั่งที่สนับสนุนการส่งข้อความลักษณะนี้แบ่งเป็นสองกลุ่มคือการส่งและรับข้อความแบบมีการยืนยัน และการส่งและรับข้อความแบบไม่มีการยืนยัน

4.5.1.1 การส่งและรับข้อความแบบมีการยืนยัน (Blocking Send and Blocking Receive)

การส่งแบบนี้จะเป็นการส่งข้อมูลที่มีความแน่นอน ขั้นตอนการส่งจะสิ้นสุดและคืนการทำงานให้กับโปรแกรมต่อเมื่อผู้รับได้รับข้อมูลเรียบร้อยแล้ว ข้อควรระวังในการใช้การส่งข้อความด้วยวิธีนี้คือการเกิดเหตุการณ์ล็อกตาย (Dead Lock) ซึ่งเป็นเหตุการณ์ที่งานรอการส่งข้อมูลตลอดไปโดยไม่มีการคืนการทำงานให้โปรแกรม ทำให้ไม่สามารถทำงานเสร็จได้ คำสั่งของ MPI ในกลุ่มคำสั่งนี้ได้แก่ MPI_SEND และ MPI_RECV

4.5.1.2 การส่งและรับข้อความแบบไม่มีการยืนยัน (Non-Blocking Send and Non-Blocking Receive)

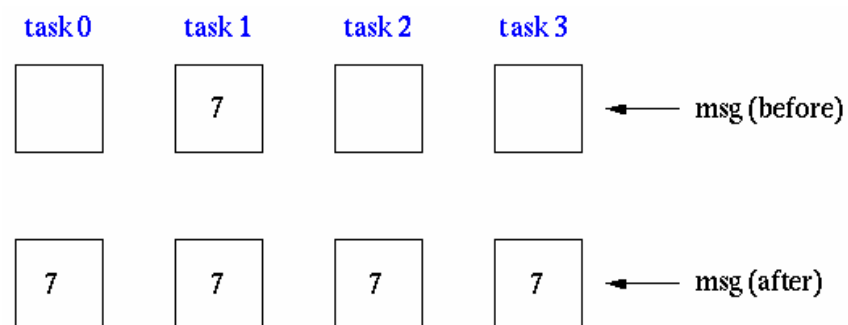
การส่งและรับข้อความแบบไม่มีการยืนยันจะถูกทำเพื่อปรับปรุงประสิทธิภาพของการส่งข้อความ การส่งข้อความแบบไม่มีการยืนยันนี้จะไม่รอให้ฝ่ายรับข้อความได้รับข้อความก่อนเหมือนการส่งข้อมูลแบบมีการยืนยันโดยจะคืนการทำงานให้โปรแกรมทันที การส่งข้อความลักษณะนี้มีความเสี่ยงในความปลอดภัยเมื่อมีการเขียนทับตัวแปรของงานที่ต้องการส่งข้อความและรับข้อความก่อนที่ข้อความจะถูกส่งไปจริง MPI จึงมีคำสั่งที่ตรวจสอบความสำเร็จของการส่งและรับด้วยคำสั่ง MPI_WAIT และ MPI_TEST เพื่อตรวจสอบสถานะการส่งและรับข้อความก่อนที่จะมี

การเขียนข้อมูลทับข้อมูลเก่า ตัวอย่างคำสั่งของการส่งและรับข้อความแบบไม่มีการยืนยันคือ MPI_ISEND และ MPI_IRECV

4.5.2 สื่อการเป็นกลุ่ม (Collective Communication)

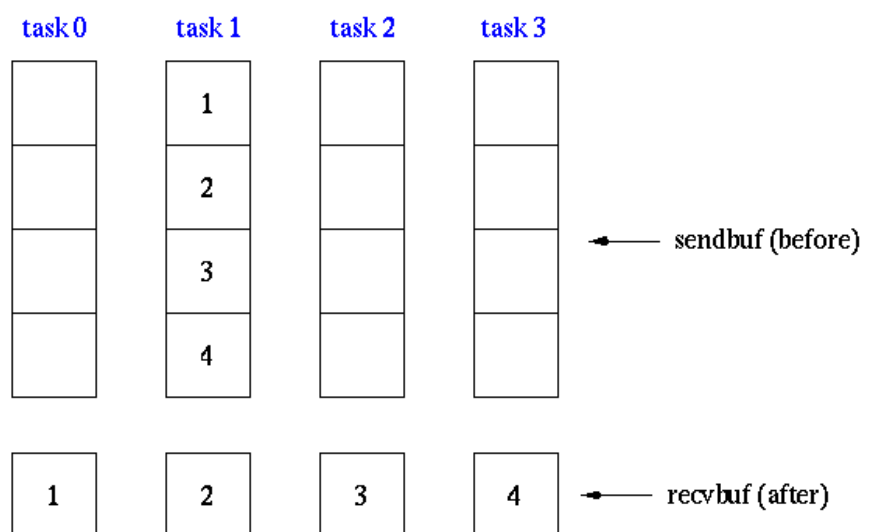
การสื่อสารแบบเป็นกลุ่มจะเป็นการสื่อสารของทุกโพรเซสในกลุ่ม ตัวอย่างคำสั่งของการสื่อสารแบบเป็นกลุ่มคือ

- MPI_Bcast เป็นคำสั่งในการกระจายค่าในตัวแปร (msg) ไปให้ทุกๆโพรเซสในกลุ่ม ดังแสดงในรูปที่ 4.9



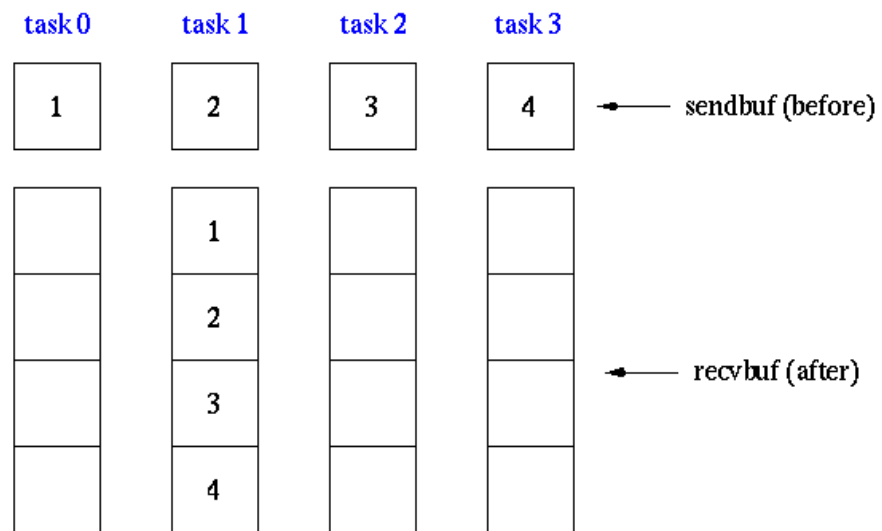
รูปที่ 4.9 ลักษณะการทำงานของ MPI_Bcast

- MPI_Scatter เป็นคำสั่งในการกระจายค่าในอะเรย์ (Array) สำหรับส่งข้อมูล (sendbuf) ไปสู่ตัวแปรของทุกๆโพรเซสในกลุ่ม (recvbuf) ตามลำดับตำแหน่งของอะเรย์ที่ใช้ส่งนั้น ดังแสดงในรูปที่ 4.10



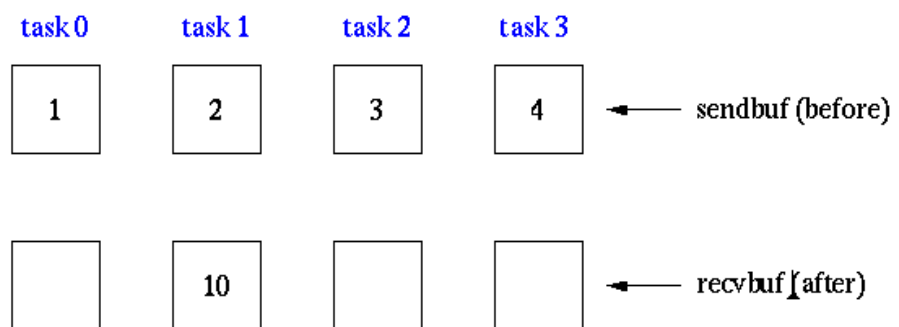
รูปที่ 4.10 ลักษณะการทำงานของ MPI_Scatter

- MPI_Gather เป็นการเก็บรวบรวมค่าในตัวแปร (sendbuf) ของโปรเซสในกลุ่มมาใส่ในอะเรย์ (recvbuf) ของโปรเซสที่เรียกใช้คำสั่งตามลำดับงานดังแสดงในรูปที่ 4.11



รูปที่ 4.11 ลักษณะการทำงานของ MPI_Gather

- MPI_Reduce เป็นการรวมค่าที่เก็บอยู่ในตัวแปร (sendbuf) ของทุกๆ โปรเซสมาเก็บในตัวแปร (recvbuf) ของโปรเซสที่เรียกใช้คำสั่งนี้ดังแสดงในรูปที่ 4.12



รูปที่ 4.12 ลักษณะการทำงานของ MPI_Reduce

4.6 สรุป

ในบทนี้ได้กล่าวถึงการจำแนกระบบคอมพิวเตอร์ การประยุกต์ใช้การประมวลผลแบบขนาน และองค์ประกอบที่จำเป็นในการสร้างระบบคลัสเตอร์ ซึ่งประกอบด้วยระบบเครือข่ายความเร็วสูงและการเชื่อมต่อแบบต่างๆ โปรแกรมสำหรับการสื่อสาร ระบบปฏิบัติการ คลัสเตอร์มีดเคิลแวร์ และโปรแกรมสร้างสถานะแวดล้อมของการโปรแกรมแบบขนาน

ในบทต่อไปจะกล่าวถึงการเขียนโปรแกรมแบบขนานลักษณะต่างๆ และการสร้างภาพตัดขวางโดยที่ใช้อัลกอริทึมเชิงพีชคณิตสำหรับวิเคราะห์ทางเดินแสงแบบกรวยด้วยระบบคลัสเตอร์โดยจะนำเสนอเป็นลำดับต่อไป