บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

สำหรับผู้ใช้ที่มีการทำงานทั่ว ๆ ไปเช่น การจัดการเอกสาร การทำบัญชี การจัดตารางเวลา ไปจนถึงเพื่อความบันเทิงนั้น คอมพิวเตอร์เพียงเครื่องเดียวก็เพียงพอที่จะตอบสนองต่อความต้องการของเขาได้อย่างมีประสิทธิภาพ แต่ในงานบางอย่างเช่น การประมวลผลภาพ (image processing) การประมวลผลอากาศพลศาสตร์ (aerodynamic processing) และการคำนวณฟิสิกส์ (physics processing) การใช้คอมพิวเตอร์เครื่องเดียวอาจทำให้ผู้ใช้ต้องรอคอยเป็นเวลานานหลายนาที หลายชั่วโมง หรือหลายสัปดาห์ ที่เป็นแบบนี้เนื่องมาจากพลังการประมวลผล (processing power) ของคอมพิวเตอร์เครื่องเดียวนั้นน้อยเกินกว่าที่จะรับมือกับงานลักษณะดังกล่าว เป็นผลให้คอมพิวเตอร์คลัสเตอร์ (computer cluster) หรือระบบการคำนวณสมรรถนะสูง (High Performnce Computing:HPC) ถูกคิดค้นขึ้นมาใช้งานจนถึงปัจจุบัน

ระบบการคำนวณสมรรถนะสูงเกือบทั้งหมดในปัจจุบันนี้ใช้สถาปัตยกรรมแบบคอมพิวเตอร์หลายเครื่อง (multicomputer architecture) หรือเรียกอีกอย่างหนึ่งว่าเบวูล์ฟคลัสเตอร์ (Beowulf cluster) โดยระบบนี้ใช้วิธีการเชื่อมคอมพิวเตอร์ (มีฮาร์ดแวร์แบบเดียวกันหรือแตกต่างกัน) เข้าด้วยกันบนเครือข่ายท้องถิ่นความเร็วสูงเช่น กิ๊กกะบิทอีเทอร์เน็ตหรืออินฟินิแบน ซึ่งผลลัพธ์ที่ตามมาจากการใช้สถาปัตยกรรมแบบคอมพิวเตอร์หลายเครื่องทำให้ต้องมีการดัดแปลงซอฟต์แวร์ดั้งเดิมที่มีการทำงานแบบลำดับ (serial) ให้เปลี่ยนเป็นแบบขนาน (parallel) เพื่อที่จะทำให้งานถูกแจกจ่ายไปบนเครื่องคอมพิวเตอร์ต่าง ๆ ในระบบประมวลผลไปพร้อม ๆ กัน ไม่เช่นนั้นสมรรถนะที่ได้ก็จะไม่แตกต่างไปจากการประมวลผลบนคอมพิวเตอร์เพียงเครื่องเดียว

เพื่อตอบสนองต่อความต้องการพลังการประมวลผลที่เพิ่มสูงขึ้นระบบเอชพีซีจึงมีแนวโน้มใหญ่โตมากขึ้นเรื่อย ๆ ทั้งในด้านอุปกรณ์การคำนวณ (computing element) และปริมาณการบริโภคพลังงาน (power consumption) ในขณะที่วิธีการใช้งานระบบเอชพีซีก็ยังคงยึดติดกับระบบซอฟต์แวร์ที่สามารถดึงพลังการประมวลผลของมันออกมาได้อย่างมีประสิทธิภาพและต้องมีเสถียรภาพในการทำงานเหมือนเดิม จึงเห็นได้ชัดเจนว่าจำนวนตัวประมวลผลที่เพิ่ม

สูงขึ้นเรื่อย ๆ เป็นอุปสรรคที่ทำให้การคำนวณของโปรแกรมประยุกต์จะถูกขัดจังหวะ (interrupt) (Schroeder & Gibson, 2007, p. 3) ดังนั้นจึงมีความพยายามศึกษากลไกการเช็คพอยต์ (checkpointing) และรีโคเวอรี่ (recovery) ขึ้นมารับมือกับสถานการณ์ดังกล่าว จากงานหลายชิ้นที่ผ่านมาแสดงให้เห็นว่าการค้นหากลไกดังกล่าวสำหรับระบบเอชพีซีไม่ใช่เรื่องง่ายนักและมีความจำเป็นต้องสร้างและดัดแปลงซอฟต์แวร์เพื่อรวมเอากลไกการทนทานต่อความผิดพร่อง (fault tolerant) เข้าไปไว้ในซอฟต์แวร์ แต่ถึงแม้วิธีการนี้จะถูกนำไปใช้ในโปรแกรมประยุกต์แบบกระจาย (distributed application) เป็นจำนวนมาก การแก้ไขซอฟต์แวร์ถูกพิจารณาว่าไม่มีความโปร่งใส (transparent) ต่อโปรแกรมประยุกต์ ทำให้โปรแกรมประยุกต์มีความซับซ้อนและใช้งานยากขึ้น

ในงานวิทยานิพนธ์ฉบับนี้เสนอวิธีการจัดเตรียมความทนทานต่อความผิดพร่องให้แก่เอชพีซีด้วยวิธีที่แตกต่างไปจากงานชิ้นอื่น ผู้วิจัยเชื่อว่าควรจัดเตรียมการทนทานต่อความผิดพร่องไว้ในซอฟต์แวร์เลเยอร์ (software layer) ที่แยกออกมาจากระบบปฏิบัติการและโปรแกรมประยุกต์ที่ทำงานอยู่ภายใน เพราะวิธีการดังกล่าวจะทำให้ระบบมีความทนทานต่อความผิดพร่องโดยไม่ต้องมีการดัดแปลงระบบปฏิบัติการและโปรแกรมใด ๆ ในการทำให้แนวคิดดังกล่าวเป็นจริงผู้วิจัยเลือกใช้เครื่องเสมือน (virtual machine) เป็นซอฟต์แวร์เลเยอร์สำหรับจัดเตรียมการทนทานต่อความผิดพร่อง ซึ่งจากการทบทวนงานประพันธ์ต่าง ๆ พบว่าเครื่องเสมือนจำนวนมากได้จัดเตรียมกลไกสำหรับเช็คพอยต์/รีโคเวอร์รีที่มีความโปร่งใสต่อระบบปฏิบัติการเกส (ระบบปฏิบัติการซึ่งถูกติดตั้งไว้บนเครื่องเสมือน) และโปรแกรมประยุกต์ไว้ให้อยู่แล้วและมีงานเพียงไม่กี่ชิ้นที่กล่าวถึงการเช็คพอยต์โปรแกรมประยุกต์แบบขนาน (parallel application) บนคลัสเตอร์ที่ประกอบขึ้นมาจากเครื่องเสมือน (Scarpazza et al., 2007 , p. 1) ถึงแม้ว่างานวิจัยไวโอลินจัดเตรียมกลไกการเช็คพอยต์/รีโคเวอร์รีที่โปร่งใสโดยแท้จริงต่อโปรแกรมประยุกต์แบบขนานในระดับของไฮเปอร์ไวเซอร์ (hypervisor – ซอฟต์แวร์เลเยอร์ที่จัดเตรียมฮาร์ดแวร์เสมือนและสามารถติดตั้งระบบปฏิบัติการลงไปได้) แต่ก็ประสบกับปัญหาแอพพลิเคชันไทม์เอาต์และการสูญหายของเฟรมข้อมูล (Kangarlou, Xu, Ruth, & Eugster, 2007, p. 4)

วิทยานิพนธ์ฉบับนี้นำเสนอซอฟต์แวร์เลเยอร์ของสถาปัตยกรรมของคลัสเตอร์เสมือน (Virtual Cluster Architecture: VCA) และระบบเช็คพอยต์คลัสเตอร์เสมือน (Virtual Cluster CheckPointing) – ซอฟต์แวร์เลเยอร์หนึ่งที่มีหน้าที่ทำการประสานงานเช็คพอยต์ (coordinator checkpointing) ให้แก่คลัสเตอร์เสมือนที่ระดับของไฮเปอร์ไวเซอร์ เสนอโพรโทคอลการเช็คพอยต์/รีโคเวอร์รีและทำการสร้างระบบขึ้นจริงบนคีมูซอฟต์แวร์ ในระหว่างการทดลองโพรโตไทป์

(prototype) กับโปรแกรมเอนพีบี (NAS Parallel Benchmark: NPB) ทำให้ผู้วิจัยได้พบกับบทเรียนต่าง ๆ และแนวทางในการวิจัยต่อในอนาคตจำนวนหนึ่ง ซึ่งสิ่งวิทยานิพนธ์ฉบับนี้นำเสนอสามารถสรุปได้ดังต่อไปนี้

1. เสนอซอฟต์แวร์เลเยอร์สำหรับคลัสเตอร์เสมือนที่ประกอบไปด้วยซอฟต์แวร์เลเยอร์ 3 ชั้นดังนี้ ชั้นเครือข่ายเสมือน (virtual network layer), ชั้นวีซีซีพี, และชั้นเครื่องเสมือน ในชั้นที่หนึ่งคือ ชั้นเครือข่ายเสมือนจำเป็นต้องมีคุณสมบัติเข้าก่อนออกก่อนและรับประกันว่าเฟรมข้อมูลจะไม่สูญหาย ส่วนชั้นที่สองคือ ชั้นวีซีซีพีซึ่งเป็นชั้นที่มีการทำเช็คพอยต์และรีโคเวอร์รีโพรโทคอล จุดสำคัญของงานชิ้นนี้ที่แตกต่างจากการเช็คพอยต์ในระดับของโพรเซสคือ วิธีการนี้ไม่ต้องตัดและสร้างการเชื่อมต่อขึ้นใหม่เมื่อมีการเช็คพอยต์หรือรีโคเวอร์รี ชั้นสุดท้ายคือ ชั้นที่สามเป็นชั้นของไฮเปอร์ไวเซอร์ เกสโอเอส และโปรแกรมประยุกต์

2. เสนอโพรโทคอลการเช็คพอยต์และรีโคเวอร์รีแบบโปร่งใสจากเกสโอเอสและโปรแกรมประยุกต์ นำเสนอระบบโพรโตไทป์ที่ใช้ทดสอบการทำงานของโพรโทคอลทั้งสอง

3. อภิปรายความถูกต้องและข้อจำกัดของโพรโทคอล แสดงให้เห็นว่าโพรโทคอลและเครือข่ายเสมือนที่ใช้ไม่ได้เป็นสาเหตุที่ทำให้เฟรมสูญหายหรือผิดลำดับ

4. แสดงให้เห็นว่าวิธีการเช็คพอยต์ที่ไฮเปอร์ไวเซอร์ของงานวิจัยนี้มีข้อจำกัดบางอย่างสำหรับโปรแกรมประยุกต์ที่ต้องการทำคล๊อกซิงโครไนเซชัน (clock synchronization) ระหว่างโพรเซสหรือมีการใช้ซอฟต์แวร์บางอย่างเพื่อตรวจจับทรานมิสชันไทม์เอาต์ (transmission timeout) อันเนื่องมาจากปัญหาที่วีเอ็มมีการหยุดและรีซูมการทำงานไม่พร้อมกัน

5. แสดงผลการวัดค่าสูญเปล่า (overhead) และสมรรถนะการเช็คพอยต์/รีโคเวอร์รีของระบบวีซีซีพี

จากผลการทดลองแสดงให้เห็นว่าการปล่อยให้โปรแกรมประยุกต์ทำงานไปตามปกติโดยไม่มีเช็คพอยต์หรือรีโคเวอร์รีนั้น เวลาที่ใช้ประมวลผลบนวีซีซีพีสูงกว่าเวลาประมวลผลบนคีมูเพียงเล็กน้อย และการทดลองนี้ก็แสดงให้เห็นอีกว่ากลไกการส่งเฟรมข้อมูลระหว่างวีเอ็มของโพรโตไทป์นี้ทำงานได้ไม่มีประสิทธิภาพ จนเป็นผลให้เวลาการประมวลผลของแนสบางเคอร์เนลที่พึ่งพาการสื่อสารสูงไม่ได้ทำงานเร็วขึ้นเมื่อจำนวนเครื่องเสมือนเพิ่มขึ้น ในด้านสมรรถนะการเช็คพอยต์พบว่าจำนวนโหนดในคลัสเตอร์มีผลต่อเวลาที่ใช้ในการเช็คพอยต์และรีโคเวอร์รี

สำหรับส่วนที่เหลือของวิทยานิพนธ์ฉบับนี้ประกอบด้วย บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง บทที่ 3 วิธีการวิจัย และสุดท้ายผลการวิจัยและการสรุปผลถูกอธิบายในบทที่ 4

1.2 วัตถุประสงค์

งานวิจัยนี้มีเป้าหมายเพื่อค้นหาและพัฒนาโพรโทคอลการเช็คพอยต์/รีโคเวอร์รีสำหรับคลัสเตอร์เสมือนที่โปร่งใสต่อระบบปฏิบัติการและโปรแกรมที่อยู่ภายในอย่างแท้จริง ดังนั้นเพื่อให้งานวิจัยดำเนินไปตามเป้าหมายดังกล่าวจึงกำหนดวัตถุประสงค์การวิจัยไว้ดังนี้

1. เพื่อเสนอเช็คพอยต์/รีโคเวอร์รีโพรโทคอลสำหรับคลัสเตอร์เสมือนที่ทำงานในระดับไฮเปอร์ไวเซอร์

2. เพื่อเสนอสถาปัตยกรรมของคลัสเตอร์เสมือนที่มีความสามารถเช็คพอยต์/รีโคเวอร์รีในระดับไฮเปอร์ไวเซอร์

3. เพื่อแสดงให้เห็นว่าโพรโทคอลที่เสนอสามารถทำงานได้โดยไม่ต้องมีการดัดแปลงระบบปฏิบัติการและโปรแกรมที่ทำงานอยู่ภายใน

4. เพื่อแสดงให้เห็นถึงประสิทธิภาพและความถูกต้องของโพรโทคอลที่เสนอ

5. เพื่อเสนอแนวทางการศึกษาวิจัยต่อเนื่อง ตลอดจนข้อเสนอแนะในการพัฒนาและปรับปรุงงานวิจัย

1.3 ขอบเขตของการวิจัย

เพื่อเป็นกรอบในการดำเนินงานวิจัยให้สามารถกรองเฉพาะประเด็นสำคัญตามวัตถุประสงค์ที่ได้กล่าวไว้ งานวิจันชิ้นนี้จึงกำหนดขอบเขตการวิจัยได้ดังต่อไปนี้

1. ทำการศึกษาระบบไฮเปอร์ไวเซอร์ของคีมูเพื่อค้นหาแนวทางการเพิ่มความสามารถเช็คพอยต์/รีโคเวอร์รีสำหรับคลัสเตอร์เสมือน

2. ทำการศึกษาและออกแบบสถาปัตยกรรมคลัสเตอร์เสมือนที่สามารถเช็คพอยต์และรีโคเวอร์รีสำหรับคลัสเตอร์เสมือนในระดับไฮเปอร์ไวเซอร์

3. ทำการศึกษาเทคนิคการเช็คพอยต์/รีโคเวอร์รีแบบต่าง ๆ เพื่อค้นหาวิธีที่เหมาะสำหรับนำมาปรับใช้ในระดับไฮเปอร์ไวเซอร์

4. ทำการศึกษาค่าสูญเปล่าที่เกิดจากการเพิ่มความสามารถเช็คพอยต์/รีโคเวอร์รี และวัดประสิทธิภาพการทำงานของโพรโทคอล

1.4 ประโยชน์ที่คาดว่าจะได้รับ

หลังจากดำเนินการวิจัยตามวัตถุประสงค์และภายใต้ขอบเขตการวิจัยที่ได้กำหนดไว้ ผู้วิจัยคาดว่าจะได้รับประโยชน์ดังนี้

1. โพรโทคอลการเช็คพอยต์/รีโคเวอร์รีแบบโปร่งใสสำหรับคลัสเตอร์เสมือน ที่สามารถเช็คพอยต์โปรแกรมประยุกต์ต่าง ๆ ได้แบบเดียวกับการเช็คพอยต์ในระดับโพรเซส

2. สถาปัตยกรรมคลัสเตอร์เสมือนที่สามารถเช็คพอยต์/รีโคเวอร์รีแบบโปร่งใสในระดับไฮเปอร์ไวเซอร์

3. โพรโทคอลการเช็คพอยต์ที่ไม่ทำให้เฟรมข้อมูลสูญหาย ลดโอกาสการเกิดแอพพลิเคชันไทม์เอาต์ และลดคล๊อกสกิวที่เกิดจากการเช็คพอยต์/รีโคเวอร์รี

4. ผลการศึกษาวิจัยและข้อเสนอแนะ เพื่อเป็นแนวทางในการศึกษาและพัฒนางานวิจัยให้มีประสิทธิภาพมากยิ่งขึ้นหรือเป็นแนวทางในการสร้างสรรค์ผลงานวิจัยด้านอื่นต่อไป

1.5 รายละเอียดโครงสร้างวิทยานิพนธ์

รายงานนำเสนอวิทยานิพนธ์ฉบับนี้แบ่งออกเป็น 4 บทได้แก่ บทนำ ทฤษฎีและงานวิจัยที่เกี่ยวข้อง วิธีการวิจัย และผลการวิจัย โดยเนื้อความในบทถัดไปมีรายละเอียดโดยสรุปดังต่อไปนี้

บทที่ 2 งานวิจัยและทฤษฎีที่เกี่ยวข้อง อธิบายงานวิจัยที่เกี่ยวข้อง ทฤษฎีการเช็คพอยต์โปรแกรมประยุกต์แบบขนาน

บทที่ 3 วิธีการวิจัย อธิบายสถาปัตยกรรมโดยรวมของคลัสเตอร์เสมือน โพรโทคอลการเช็คพอยต์/รีโคเวอร์รี ระบบวีซีซีพี ความถูกต้องและข้อจำกัด

บทที่ 4 ผลการวิจัย อธิบายถึงเครื่องที่ใช้ทำการวิจัย แสดงผลการทดลองโอเวอร์เฮดสมรรถนะการเช็คพอยต์/รีโคเวอร์รี การสรุปผลและงานในอนาคต