บทที่ 6

บทสรุป

6.1 สรุปผลหลัก

ส่วนประกอบทางฮาร์ดแวร์ที่ทำหน้าที่เก็บข้อมูลในคอมพิวเตอร์คือหน่วยความจำหลัก ซึ่งเป็นหน่วยความจำแบบ RAM(Random Access Memory) เป็นหน่วยความจำที่หน่วยประมวลผลกลางเข้าถึงข้อมูลได้โดยตรงใช้เวลาในการเข้าถึงข้อมูลน้อย กับหน่วยความจำรองซึ่งเป็นหน่วยความจำ DASD(Direct Access Storage Devices) เป็นหน่วยความจำที่หน่วยประมวลผลกลางเข้าถึงข้อมูลได้โดยผ่านหน่วยรับส่งข้อมูล ใช้เวลาในการเข้าถึงข้อมูลมากแต่มีความจุข้อมูลมากกว่าหน่วยความจำ RAM

ระบบหน่วยความจำลำดับชั้น เป็นระบบหน่วยความจำที่ประกอบด้วยหน่วยความจำตั้งแต่สองชุดที่มีความจุข้อมูลและเวลาในการเข้าถึงข้อมูลแตกต่างกันมาประกอบกันเป็นลำดับชั้น เพื่อให้ได้ระบบหน่วยความจำที่มี ความจุข้อมูล, เวลาในการเข้าถึงข้อมูล, และราคาต่อหน่วยความจุข้อมูลที่เหมาะสม หน่วยความจำเสมือน, หน่วยความจำแคช, และบัฟเฟอร์แคช เป็นระบบหน่วยความจำลำดับชั้น หน่วยความจำเสมือนเป็นระบบหน่วยความจำลำดับชั้นที่มีจุดมุ่งหมายเพื่อขยาย Logical Address Space หน่วยความจำแคชเป็นระบบหน่วยความจำลำดับชั้นที่ช่วยให้การส่งผ่านข้อมูลระหว่างหน่วยประมวลผลกับหน่วยความจำหลักดีขึ้น บัฟเฟอร์แคชเป็นระบบหน่วยความจำลำดับชั้นที่ช่วยให้การส่งผ่านข้อมูลระหว่างหน่วยประมวลผลกับหน่วยความจำรองดีขึ้น

โปรแกรมควบคุมบัฟเฟอร์แคชที่ทดลองสร้างขึ้นมีการทำงานแบบ Write-through ทำการติดตั้งกับ Disk-BIOS โดยผ่านคำสั่ง DOS เช่นเดียวกับโปรแกรม Terminate and Stay Resident ทั่วๆไป

การทดสอบบัฟเฟอร์แคชที่สร้างขึ้นมี 2 ส่วนคือ การทดสอบสมรรถนะการทำงานของบัฟเฟอร์แคชและการทดสอบการทำงานของโปรแกรมประยุกต์เมื่อใช้บัฟเฟอร์แคช การทดสอบสมรรถนะการทำงานของบัฟเฟอร์แคชมีจุดประสงค์เพื่อทดสอบบัฟเฟอร์แคชที่สร้างขึ้นช่วยให้การอ่านข้อมูล

ในดิสค์ดีขึ้นอย่างไรและมีอะไรเป็นตัวแปรที่มีผลกับการทำงานของบัฟเฟอร์แคชบ้าง การทดสอบสมรรถนะการทำงานของบัฟเฟอร์แคช ทำโดยโปรแกรมทดสอบซึ่งทำการอ่านข้อมูลจากดิสค์ผ่าน Disk-BIOS ตามแบบอ้างอิง(Reference Pattern)

การทดสอบการทำงานของโปรแกรมประยุกต์เมื่อใช้บัฟเฟอร์แคชมีจุดประสงค์เพื่อแสดงให้เห็นว่าบัฟเฟอร์แคชช่วยให้การทำงานของโปรแกรมประยุกต์ดีขึ้น โปรแกรมประยุกต์ที่ใช้ในการทดสอบคือโปรแกรมจัดการฐานข้อมูล dBase III plus โดยกำหนดไฟล์ข้อมูลขึ้นมา 1 ไฟล์เพื่อให้โปรแกรมจัดการฐานข้อมูล dbase III plusทำงานกับข้อมูลในไฟล์ข้อมูลนั้น

จากผลการทดสอบการทำงานของบัฟเฟอร์แคชทั้งกับแบบอ้างอิงแบบลำดับวนและกับแบบอ้างอิงแบบสุ่ม จะเห็นว่าเมื่อบัฟเฟอร์แคชมีจำนวนบัฟเฟอร์บล็อกเท่ากับหรือมากกว่าจำนวนบล็อกของกลุ่มข้อมูลที่ถูกอ้างอิงโดยแบบอ้างอิงในขณะนั้น การอ่านข้อมูลในดิสค์จะดีขึ้นมากเนื่องจากข้อมูลทั้งหมดที่แบบอ้างอิงอ้างถึงอยู่ในบัฟเฟอร์แคชทั้งหมด

เมื่อจำนวนบัฟเฟอร์บล็อกน้อยกว่าจำนวนบล็อกของกลุ่มข้อมูล ในแบบอ้างอิงสุ่มบัฟเฟอร์แคชช่วยให้การอ่านข้อมูลในดิสค์ดีขึ้นมาบ้างตามอัตราส่วนระหว่างจำนวนบัฟเฟอร์บล็อกกับจำนวนบล็อกของกลุ่มข้อมูลดังนี้

$$H = \frac{N_b}{N_d} \qquad (6.1)$$

$$t_a = Ht_{a1} + (1 - H)t_{a2} \qquad (6.2)$$

N_b = จำนวนบัฟเฟอร์บล็อก

N_d = จำนวนบล็อกของข้อมูล

H = Hit Ratio

t_a = เวลาเข้าถึงข้อมูลเฉลี่ย

t_{a1} = เวลาเข้าถึงข้อมูลที่มีอยู่ในบัฟเฟอร์

t_{a2} = เวลาเข้าถึงข้อมูลที่ไม่มีในบัฟเฟอร์

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ส่วนในการอ้างอิงข้อมูลแบบลำดับวนนั้น เมื่อจำนวนบัฟเฟอร์บล็อกน้อยกว่าจำนวนบล็อกของกลุ่มข้อมูลที่ถูกอ้างอิง บัฟเฟอร์แคชจะไม่ช่วยในการอ่านข้อมูลเลย

ในขณะที่โปรแกรมประยุกต์ทำงานนั้น การอ้างอิงข้อมูลที่เกิดขึ้นไม่ได้เป็นการอ้างอิงแบบลำดับวนหรือแบบสุ่มอย่างใดอย่างหนึ่ง แต่จะเป็นได้ทั้งสองอย่างดังเช่นการทำงานของโปรแกรมจัดการฐานข้อมูล dBase III pluse ที่ใช้ในการทดสอบการทำงานในการอ่านข้อมูล และการเขียนข้อมูลเรียงตั้งแต่ Record แรกถึง Record สุดท้าย ทำให้เกิดการอ้างอิงข้อมูลแบบลำดับวน แต่ถ้าเป็นการอ่านข้อมูลและการเขียนข้อมูลโดยใช้ไฟล์ดัชนีช่วยจะเกิดการอ้างอิงข้อมูลที่มีลักษณะใกล้เคียงการอ้างอิงแบบสุ่ม บัฟเฟอร์แคชมีส่วนช่วยให้การทำงานของโปรแกรมประยุกต์ดีขึ้นบ้างในกรณีที่จำนวนบัฟเฟอร์บล็อกน้อยกว่าที่จะเก็บข้อมูลในไฟล์ได้หมด โดยทั่วๆไประบบฐานข้อมูลจะประกอบด้วยไฟล์ข้อมูลหลายๆไฟล์ แต่เมื่อโปรแกรมประยุกต์ทำงานกับฐานข้อมูล โปรแกรมประยุกต์โดยทั่วไปจะไม่ใช้ไฟล์ข้อมูลทุกๆไฟล์แต่จะใช้ไฟล์ข้อมูลบางไฟล์เท่านั้นทำให้เกิดการอ้างอิงข้อมูลเป็นกลุ่มๆขึ้นซึ่งในกรณีเช่นนี้ บัฟเฟอร์แคชควรมีบัฟเฟอร์บล็อกมากพอเก็บข้อมูลทั้งกลุ่มในขณะนั้นได้

6.2 อุปสรรค์ในการพัฒนาบัฟเฟอร์แคช

ระบบปฏิบัติการ MS-DOS เป็นระบบปฏิบัติการที่เริ่มแรกพัฒนาขึ้นด้วยภาษา Assembly การส่งผ่านค่าพารามิเตอร์จะส่งผ่านรีจิสเตอร์รวมไปถึงการส่งผ่านพารามิเตอร์ระหว่างระบบปฏิบัติการ MS-DOS กับ BIOS ด้วย ในปัจจุบันนี้ ภาษา C เป็นภาษาที่เข้ามามีบทบาทสูงในการพัฒนาซอฟแวร์เชิงระบบ ภาษา C มีการส่งผ่านข้อมูลบน Stack และภาษา C ยังจัดสรรหน่วยความจำใน Stack ให้เป็นตัวแปรอีกทำให้รูทีนที่เขียนขึ้นด้วยภาษา C จะต้องการพื้นที่สำหรับ Stack มาก ในขณะที่ระบบปฏิบัติการ MS-DOS จัดพื้นที่หน่วยความจำให้เพียงเล็กน้อยจึงเป็นเหตุผลสำคัญที่ต้องใช้ภาษา Assembly ในการสร้างบัฟเฟอร์แคชในครั้งนี้

นอกจากนี้แล้ว ระบบปฏิบัติการ MS-DOS มีการทำงานแบบงานเดียว(Single Task) เมื่อระบบไฟล์(File System)ซึ่งเป็นส่วนหนึ่งของระบบปฏิบัติการต้องการอ่าน/เขียนข้อมูลในดิสค์

ระบบปฏิบัติการจะรอผลการทำงานของ Disk-BIOS ในการดำเนินการอ่าน/เขียนข้อมูล นี้เป็นจุดหนึ่งที่เป็นอุปสรรค์ในการสร้างบัฟเฟอร์แคชแบบ Write Cache ที่สมบูรณ์โดยการสร้างรูทีนขึ้นเสริมการทำงานดังที่ใช้มาตั้งแต่ต้นได้ เพราะเมื่อระบบปฏิบัติการต้องการอ่านข้อมูลจากดิสค์ถ้าบัฟเฟอร์แคชไม่มีข้อมูล บัฟเฟอร์แคชจะอ่านข้อมูลจากดิสค์ซึ่งถ้าการอ่านข้อมูลจากดิสค์ไม่ประสบผลสำเร็จบัฟเฟอร์แคชจะแจ้งให้ระบบปฏิบัติการทราบ ในการเขียนข้อมูล ถ้าบัฟเฟอร์แคชทำงานแบบ Write Cache เมื่อระบบปฏิบัติการเขียนข้อมูลลงดิสค์ บัฟเฟอร์แคชต้องหลอกระบบปฏิบัติการว่าได้เขียนข้อมูลลงดิสค์แล้ว เมื่อถึงเวลาที่บัฟเฟอร์แคชต้องเขียนข้อมูลลงดิสค์จริงๆแล้วไม่สามารถเขียนข้อมูลลงดิสค์ได้ ข้อมูลนั้นจะสูญหาย

6.3 แนวทางที่จะนำบัฟเฟอร์แคชไปประยุกต์ใช้งานอื่นๆ

บัฟเฟอร์แคช เป็นวิธีการประยุกต์หน่วยความจำมาช่วยให้การส่งผ่านข้อมูลระหว่างดิสค์กับระบบปฏิบัติการดีขึ้น หลักการของบัฟเฟอร์แคชนอกจากประยุกต์ใช้กับการส่งผ่านข้อมูลระหว่างดิสค์กับระบบปฏิบัติการแล้ว อาจนำไปประยุกต์ใช้ในการส่งผ่านข้อมูลระหว่างระบบปฏิบัติการกับโปรแกรมประยุกต์ได้โดยใช้บัฟเฟอร์แคชเป็นรูทีนย่อยๆที่รวมอยู่ในโปรแกรมประยุกต์ และอาจนำมาประยุกต์ใช้ในการส่งผ่านข้อมูลผ่านระบบโทรคมนาคม เช่นการส่งผ่านข้อมูลระหว่างตัวบริการไฟล์(File Server)กับเครื่องไมโครคอมพิวเตอร์ที่ขอใช้ข้อมูลจากตัวบริการไฟล์ผ่านระบบเครือข่ายท้องถิ่น(Local Area Network)ได้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย