

บทที่ 3

แฟ้มข้อมูล

3.1 ความหมายของแฟ้มข้อมูล

โดยปกติแฟ้มข้อมูลจะถูกนิยามให้เป็นกลุ่มของสารสนเทศที่มีความสัมพันธ์กัน โดยไม่มีรูปลักษณ์หรือโครงสร้างใดๆ นอกเหนือไปจากการแบ่งซอยเป็นระเบียบเท่านั้น (เปรียบเสมือนกับแฟ้มกระดาษที่แบ่งแยกเป็นหน้า หรือเป็นแผ่นๆ ตามความสะดวกของการจัดเก็บและใช้งาน) การไร้รูปลักษณ์นี้ไม่คำนึงถึงลักษณะหรือโครงสร้างของอุปกรณ์ที่ใช้เก็บ ประโยชน์ประการหนึ่งของหลักการของแฟ้มเสมือนไร้ลักษณะ (formatless virtual file) คือ เป็นสื่อกลางในการเชื่อมโยงหลักการของแฟ้มตรรกและแฟ้มกายภาพเข้าด้วยกัน แฟ้มตรรก (logical file) หมายถึงแฟ้มงานแ่งมุมของผู้ใช้ ซึ่งมีโครงสร้างเป็นไปตามตรรกการดำเนินงานของโปรแกรมประยุกต์ ในขณะที่แฟ้มกายภาพ (physical file) คือ แฟ้มที่ปรากฏอยู่บนสื่อบันทึกข้อมูล ซึ่งมีโครงสร้างไปตามลักษณะของอุปกรณ์นั้นๆ โครงสร้างทั้งสองแบบนี้อาจสมนัยกันหรือไม่ก็ได้ ในสมัยแรกเริ่มหรืองานประยุกต์ที่ซับซ้อนจะพยายามออกแบบแฟ้มข้อมูลให้มีโครงสร้างทางตรรกและกายภาพที่เหมือนกัน หรือเหมาะสมกัน เพื่อประสิทธิภาพในการใช้งานที่ดี แต่ในปัจจุบันจะคำนึงถึงความสะดวกของผู้ใช้เป็นหลัก ดังนั้นจึงมีแรงผลักดันให้แยกโครงสร้างทางตรรกออกจากโครงสร้างทางกายภาพ เพื่อให้ผู้ใช้จะไม่ต้องพะวงกับลักษณะโครงสร้างในอุปกรณ์

หลักการแฟ้มข้อมูลเสมือนไร้ลักษณะ สามารถใช้แบ่งแยกหลักการทางตรรกและทางกายภาพได้ดี ดังจะเห็นได้ในรูปที่ 3.1 การที่แฟ้มเสมือนไร้ลักษณะไม่มีโครงสร้างหรือรูปแบบใดๆ นอกเหนือไปจากลักษณะการเป็นแถวอักษร ทาให้เป็นโครงสร้างที่เป็นกลางและยืดหยุ่นมาก สามารถแปลง เปลี่ยนเป็นโครงสร้างใดๆ ก็ได้ ไม่ว่าจะเป็นทางตรรกหรือทางกายภาพ

จะเห็นว่าตามตัวอย่างในรูปที่ 3.1 นั้น ขนาดของระเบียบในแฟ้มกายภาพไม่สมนัยกับขนาดของระเบียบในแฟ้มตรรก ซึ่งตามหลักการพื้นฐานของการประมวลผลข้อมูล จะทำให้ประสิทธิภาพในการเข้าถึงข้อมูลต่ำลง ดังนั้นโดยทั่วไปข้อมูลจะได้รับการออกแบบโดยอาศัยโครงสร้างทางกายภาพเป็นหลัก ซึ่งทำให้มองข้ามโครงสร้างไว้ลักษณะใบโคยสิ้นเชิง แต่ปัจจุบันนี้ระบบปฏิบัติการรุ่นใหม่ เช่น ยูนิกซ์ หรือระบบปฏิบัติการสำหรับเครื่องไมโครคอมพิวเตอร์ ได้เริ่มหันมาชมรมทัศน์ของโครงสร้างไว้ลักษณะเพื่อลดปัญหาการไม่เข้ากันของอุปกรณ์ต่างๆ ที่ประกอบอยู่ในระบบ เช่น งานแม่เหล็กต่างยี่ห้อกัน

แฟ้มตรรก	:	1612	:	นายสกลนคร	ชาญการ	:
				:	1674	:
				:	น.ส.วรรณภา	สายสมร
				:	----	
				:	----	
				:	----	
แฟ้มเสมือน		1612		นายสกลนคร_ชาญการ		1674
						น.ส.วรรณภา_สายสมร
					
						ไว้ลักษณะ
						(แถวของอักขระ)
แฟ้มกายภาพ:		1612		นายสกลนคร	ชา	:
						(สมมติให้แต่ละบล็อกยาว 16 อักขระ)
				:	ญการ	1674
				:	น.ส.:	
				:	วรรณภา	สายสมร
				:	----	
				:	----	
				:	----	

รูปที่ 3.1 รูปลักษณะของแฟ้มแบบต่างๆ

3.2 กรรมวิธีของการประมวลผลกับแฟ้มข้อมูล

กรรมวิธีที่ใช้ในการประมวลผลกับแฟ้มข้อมูลนั้นจะมีกรรมวิธีที่เกี่ยวข้องอยู่หลายประการ เช่นการ เรียงลำดับข้อมูล(sort) การรวมข้อมูล (merge) การจับคู่ระเบียบข้อมูล (match) เป็นต้น ซึ่งวิธีการดำเนินการกับแฟ้มข้อมูลสามารถแยกตามประเภทของงานได้เป็น ประเภทที่ดำเนินการอยู่บ่อยๆโดยทั่วไป กับ ประเภทที่เน้นเฉพาะงานด้านที่นำไปประยุกต์ใช้ ในที่นี้จะกล่าวเฉพาะประเภทที่ดำเนินการอยู่บ่อยๆโดยทั่วไปซึ่งมีดังต่อไปนี้

3.2.1 การค้นหาข้อมูล คือ การค้นหาค่าแห่งของระเบียบข้อมูลที่อยู่ในแฟ้มข้อมูลที่ตรงกับเงื่อนไขที่เราต้องการ กรรมวิธีการค้นหานั้นก็อาจจะทำได้หลายแบบ เช่น ตั้งค้นค้นหา ตั้งแต่ระเบียบข้อมูลแรกและ เรื่อยๆไปที่ละหน่วยระเบียบข้อมูลตามลำดับที่ตั้งในแฟ้มข้อมูล จนกว่าจะพบสิ่งที่ต้องการ หรืออาจจะใช้กรรมวิธีที่ซับซ้อนขึ้นไปอีก ซึ่งอาจมีจุดประสงค์เพื่อดำเนินการปรับแก้ข้อมูล หรือลบระเบียบข้อมูล หรือการแทรกเพิ่มเติมระเบียบข้อมูลเข้าไป การดำเนินการค้นหาข้อมูลนั้นมีจุดมุ่งหมายอยู่ 2 ประการคือ ประการที่หนึ่งค้นหาว่าข้อมูลที่เราต้องการนั้นปรากฏอยู่ในแฟ้มข้อมูลหรือไม่ และ ประการที่สองก็เพื่อหาค่าแห่งของข้อมูลที่ต้องการว่าปรากฏอยู่ที่ตำแหน่งใดในแฟ้มข้อมูล

3.2.2 การค้นคืนข้อมูล คือ กรรมวิธีในการดึงระเบียบข้อมูลให้เข้าไปเก็บในหน่วยความจำหลัก เพื่อจะประมวลผลต่อไป ในกรณีที่แฟ้มข้อมูลนั้นปรากฏอยู่ในหน่วยความจำหลักอยู่แล้ว จะหมายถึงการค้นหาตำแหน่งของระเบียบข้อมูลที่ปรากฏในขณะนั้นว่าอยู่ตำแหน่งไหนในแฟ้มข้อมูล

3.2.3 การปรับปรุงข้อมูล คือ การเปลี่ยนแปลงข้อมูลในแฟ้ม โดยปกติจะหมายถึง การปรับแก้รายการข้อมูลให้ตรงกับเหตุการณ์ในขณะนั้น การปรับปรุงข้อมูลมีอยู่ 2 ประเภท คือ

- การเปลี่ยนแปลงโครงสร้างของแฟ้มข้อมูล (หมายถึงการเพิ่มและตัดทอนบางระเบียบข้อมูล) และ เปลี่ยนแปลงแก้ไข เฉพาะบางรายการข้อมูลของระเบียบข้อมูลเท่านั้น
- การ posting หน่วยระเบียบข้อมูล คือการส่งค่า (value) ให้เข้าไปเก็บยังตำแหน่งของรายการข้อมูลในหน่วยระเบียบข้อมูลนั้น ซึ่งกรรมวิธีที่ว่านี้ค่าที่ส่งเข้าไปเก็บจะไม่มีผลกระทบต่อโครงสร้างของแฟ้มข้อมูล กรรมวิธีนี้อาจจะหมายถึงการเริ่มกำหนดค่าของรายการข้อมูล หรืออาจจะเป็นการเปลี่ยนแปลงมูลค่าเดิมที่มีอยู่แล้วในรายการข้อมูลก็ได้

สาเหตุที่เน้นเฉพาะการ posting กับรายการข้อมูล ซึ่งไม่ก่อให้เกิดผลกระทบกับโครงสร้างของแฟ้มข้อมูล ก็เพราะว่าโครงสร้างของแฟ้มข้อมูลโดยปกติจะขึ้นอยู่กับมูลค่าที่ปรากฏในเขตข้อมูลหลัก (key field) ถ้าหากมีการเปลี่ยนแปลงข้อมูลในเขตข้อมูลหลักแล้วจะรวมถึงมีการลบและเพิ่มระเบียนข้อมูลในแฟ้มข้อมูลด้วย ตัวอย่างเช่น ถ้าเรามีแฟ้มข้อมูลของพนักงานอยู่แฟ้มหนึ่ง ในแฟ้มข้อมูลนี้จะประกอบด้วยระเบียนข้อมูลของพนักงานแต่ละคน ซึ่งจะอยู่เรียงลำดับจากน้อยไปหามากตามชื่อของพนักงาน ในกรณีนี้เราเกิดเปลี่ยนแปลงชื่อของพนักงาน เราก็จำเป็นต้องเรียงลำดับเสียใหม่ นั่นก็คือโครงสร้างแฟ้มข้อมูลนั้นจะต้องนำไปจัดเรียงลำดับใหม่ จะเห็นได้ว่าถ้าหากเปลี่ยนแปลงชื่อเดิม TAME S, R เป็นชื่อใหม่ JAMES, R แล้วก็จำเป็นต้องเรียงลำดับระเบียนข้อมูลในแฟ้มข้อมูลเสียใหม่ ทั้งนี้เพื่อที่แฟ้มข้อมูลนั้นจะยังมีลักษณะของการเรียงลำดับจากน้อยไปหามาก ผลจากการเปลี่ยนแปลงนี้จะปรากฏดังรูปที่ 3.2



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Location	Key	Other Fields
1	ADAMS, B	
2	BAKER, J	
3	DURAN, A	
4	MORRIS, W	
5	SMITH, F	
6	TAMES, R	

Location	Key	Other Fields
1	ADAMS, B	
2	BAKER, J	
3	DURAN, A	
4	JAMES, R	
5	MORRIS, W	
6	SMITH, F	

รูปที่ 3.2 แสดงการเปลี่ยนแปลงของแฟ้มข้อมูลเมื่อเปลี่ยนเขตข้อมูลหลัก

3.3 วิธีการเข้าถึงข้อมูล

ในแง่มุมมองของระบบเครื่องคอมพิวเตอร์ แฟ้มข้อมูลอาจเก็บอยู่ในรูปของบล็อกกระจายอยู่ในจานแม่เหล็กบันทึกในลักษณะของแฟ้มทางกายภาพหรืออาจเป็นแถวลำดับของหน่วยสารสนเทศที่เรียงต่อกันไปไบต์ต่อบิตในลักษณะของแฟ้มเสมือนไร้ลักษณะ แต่ในแง่มุมมองของผู้ใช้สารสนเทศ แฟ้มข้อมูลมีลักษณะการจัดเรียงที่เหมาะสม หรือ สมนัยกับการประยุกต์สารสนเทศนั้น การใช้งานหรือการเข้าถึง (access) แฟ้มเชิงตรรก เหล่านี้ก็ซับซ้อนตามโครงสร้างของแฟ้มไปด้วย ซึ่งบ่อยครั้งที่เป็นการเข้าถึงระเบียบในแฟ้มจัดการระบุเขตข้อมูลหลักบาง เขตของระเบียบ วิธีการเข้าถึง (access method) เป็นกลไกของระบบที่ช่วยอำนวยความสะดวกการใช้แฟ้มในระดับของผู้ใช้ วัตถุประสงค์การจัดการแฟ้มเชิงตรรก (เช่นการสอคแทรกระเบียบ) รวมทั้งการแปลงความต้องการหรือการกระทำต่างๆให้อยู่ในรูปที่เหมาะสมต่อระบบแฟ้มกายภาพ (เพราะต้องไปอ่าน/บันทึกข้อมูลกับแฟ้มกายภาพในที่สุดอยู่ดี)

วิธีการเข้าถึงข้อมูลที่เข้ากันอยู่นั้นมีจำนวนมากและแต่ละวิธีการก็ซับซ้อนซึ่งในที่นี้ จะกล่าวเฉพาะหลักการที่สำคัญเกี่ยวข้องกับการเข้าถึงในระดับของผู้ใช้โดยวิธีการเข้าถึงในทางหลักการนี้ แบ่งได้เป็น 2 ประเภทคร่าวๆ ดังนี้

3.3.1 การเข้าถึงโดยอาศัยตำแหน่ง สำหรับวิธีการเข้าถึงข้อมูลประเภทนี้พิจารณาเฉพาะช่องหรือบริเวณที่บรรจุข้อมูล โดยไม่สนใจกับเนื้อความที่มีอยู่ในบริเวณนั้นๆ เปรียบเสมือนการพิจารณาเฉพาะที่ตั้งของบ้านพักอาศัยโดยมิได้สนใจว่าเป็นบ้านชั้นเดียวหรือสองชั้น มีบริเวณหรือไม่ เจ้าของเป็นใคร มีใครอยู่บ้าง จะสนใจก็เฉพาะบริเวณหรือขนาดของที่ดินเท่านั้น ซึ่งขนาดของที่ดินมีผลต่อการอ้างอิงตำแหน่งที่ตั้ง เชิงสัมพัทธ์

วิธีการเข้าถึงโดยอาศัยตำแหน่งนี้ แบ่งออก เป็นสองประเภทคือ

ก) วิธีการเข้าถึงแบบเรียงลำดับ (sequential access method)

เมื่อกล่าวถึงวิธีการเข้าถึงแบบเรียงลำดับ ก็มักนึกถึงแฟ้มข้อมูลที่อยู่ในเทปแม่เหล็กนั่นคือการเข้าถึงนอกจากจะเป็นโดยตำแหน่งโดยไม่สนใจเนื้อความแล้วยังเป็นตำแหน่งสัมพัทธ์อย่างง่ายด้วย กล่าวคือสามารถเข้าถึงข้อมูลในระเบียบหรือไบต์ "ก่อน"(previous) หรือ "หลัง"(next) ข้อมูลที่ตำแหน่ง"ปัจจุบัน"(current) จะเห็นว่าในการประมวลผลเรียง

ลำดับ (sequential processing) ที่ใช้กันโดยปกตินั้นก็เป็นการเข้าถึงโดยอาศัยตำแหน่งแบบสัมพัทธ์และ เฉพาะตำแหน่งหลังจากตำแหน่งปัจจุบันเท่านั้น

ข) วิธีการเข้าถึงแบบตรง (direct access method)

ในการเข้าถึงแบบนี้ใช้ตำแหน่งแห่งที่เฉพาะของสารสนเทศ (ในระดับระเบียบ หรือ ระดับไบต์) ซึ่งตำแหน่งเหล่านี้ปกติเป็นเลขจำนวนเต็มเรียงกันจากน้อยไปมาก โดยเริ่มนับตำแหน่งที่ค้นเพิ่มเป็นตำแหน่งแรก ซึ่งแน่นอนว่าการจัดเก็บแฟ้มต้องเอื้ออำนวยต่อการเข้าถึงแบบนี้ด้วย หากการจัดเก็บข้อมูลมีความจำที่เป็นแบบเรียงลำดับ เช่น เทป ก็ต้องมีกลไกซึ่งจำลองการทำงานแบบตรงด้วยการทำงานแบบเรียงลำดับ เช่น ใช้วิธีอ่านเรียงลำดับไบต์และระเบียบ พร้อมกับนับจำนวนระเบียบที่อ่านไปจนกระทั่งถึงระเบียบหมายเลขที่ต้องการ เป็นต้น การเข้าถึงแบบตรงนี้อาจระบุตำแหน่งความหน่วยย่อยที่สุดของสารสนเทศ เช่น เป็นตำแหน่งของไบต์ นับจากต้นแฟ้มหรืออาจระบุเป็นหมายเลขระเบียบ ซึ่งในกรณีหลังต้องมีการแปลงหมายเลขระเบียบออกเป็นตำแหน่งไบต์ เพื่อให้สามารถเข้าถึงตำแหน่งที่ระบุไว้ได้ (ซึ่งยังน่าจะเป็นการเข้าถึงในงานบันทึก) การแปลงนี้สามารถกระทำได้ง่ายเมื่อขนาดของระเบียบคงที่ (fixed-length record)

วิธีการเข้าถึงข้อมูลแบบตรงนี้ ยึดหยุ่นกว่าแบบเรียงลำดับและถูกใช้เป็นที่รากฐานสำหรับวิธีการเข้าถึงแบบอื่น ๆ ซึ่งอาศัยเนื้อความ ดังจะกล่าวถึงต่อไป

3.3.2 การเข้าถึงโดยอาศัยเนื้อความ ในวิธีการเข้าถึงโดยอาศัยเนื้อความนี้ ผู้ใช้สามารถระบุสารสนเทศที่ต้องการโดยอาศัยเนื้อความหรือสารสนเทศที่บรรจุอยู่ในแฟ้มนั้นแทนการใช้ตำแหน่ง ในการระบุโดยเนื้อความนี้สารสนเทศมักถูกรวบรวมเป็นหน่วยใหญ่ เช่น เป็นระเบียบแล้วอาศัยบางส่วนของระเบียบนี้ (คือเซค) ในการระบุสารสนเทศในระเบียบ เซคที่ใช้ในการระบุนี้เรียกว่า กุญแจ (key) ซึ่งอาจเป็นกุญแจเฉพาะ (unique key) ที่แยกแยะระเบียบจากกันโดยเด่นชัด เช่นรหัสประจำตัวพนักงาน หรืออาจเป็นกุญแจบางส่วน (partial key) ซึ่งระบุระเบียบได้มากกว่าหนึ่งระเบียบ เช่นอัตราเงินเดือนของพนักงาน

แม้จะสามารถระบุสารสนเทศกลุ่มที่ต้องการด้วยเนื้อความหรือกุญแจ แต่การเข้าถึงก็ยังคงอาศัยตำแหน่งอยู่ดี (สำหรับอุปกรณ์ความจำทั่วไป) ดังนั้นต้องมีกลไกการแปลงเนื้อความมาเป็นตำแหน่ง ซึ่งอาจเป็นแบบที่ซับซ้อนอย่างไรก็ได้ เช่นหากใช้การจัดเรียงแฟ้มข้อมูลบนอุปกรณ์

ประเภทเรียงลำดับก็ใช้วิธีอ่านเรียงลำดับไปเรื่อยๆ จากต้นแฟ้มจนพบระเบียบที่มีเนื้อความของ
คุณเจตรงกับเนื้อความที่ต้องการ

สำหรับการใช้เนื้อความร่วมกับระบบการเข้าถึงโดยตรงนั้น อาจใช้วิธีการแปลง
เนื้อความเป็นตำแหน่งแบบง่าย ๆ โดยทำเป็นตารางว่าเนื้อความใดอยู่ที่ตำแหน่งใดในแฟ้ม
ตารางนี้เรียกว่าแฟ้มผกผัน (inverted file) หรือ คัดชนี (index) อีกวิธีการคือการแปลง
โดยใช้ขั้นตอนทางคณิตศาสตร์ หรือ สูตรสำเร็จ เช่น การนำเอาอักษรในลักษณะรหัสเลข
ฐานสองของเนื้อความมาบวกกันเข้าหาค่าออกเป็นตำแหน่งในแฟ้ม ลักษณะการแปลงเช่นนี้
เรียกว่า แฮชชิ่ง (hashing) หรือในอีกวิธีการแปลงที่เรียกว่าคัตชนี เรียงลำดับ (indexed
sequential) เป็นการผสมกันระหว่างการเข้าถึงตารางและการเข้าถึงโดยใช้ตำแหน่งเชิงสัมพัทธ์
กล่าวคือคัดแบ่งสารสนเทศในแฟ้มให้เป็นบล็อกใหญ่กว่า ระเบียบระเบียบ (แต่ละบล็อกมีมากกว่าหนึ่ง
ระเบียบ) แล้วเก็บคัตชนีเฉพาะระเบียบสุดท้ายในบล็อก การเข้าถึงโดยเนื้อความก็จะค้นดูว่า
ระเบียบที่ต้องการคอกอยู่ในบล็อกใดแล้วจึงค้นแบบเรียงลำดับในบล็อกนั้นอีกทีหนึ่ง

3.4 การรักษาโครงสร้างของแฟ้มข้อมูลแบบเรียงลำดับ

เมื่อใดก็ตามที่มีการสร้างแฟ้มข้อมูลมาเพื่อนำมาใช้ งาน เราก็จำเป็นต้องตรวจสอบ
ให้แน่ใจว่าข้อมูลที่ปรากฏในแฟ้มข้อมูลนั้นถูก เก็บอย่างครบถ้วน รวมทั้งที่โครงสร้างและการ
เรียงลำดับอย่างที่เรากำหนดเอาไว้ สิ่งทั้งหลายที่เราต้องการนี้จะรวมอยู่ในกิจกรรมที่เราจะ
ต้องดำเนินการกับแฟ้มข้อมูลซึ่งกิจกรรมเหล่านี้ เราวมเรียกว่า file maintenance
โดยปกติทั่วไปถ้าเราดำเนินการกับแฟ้มข้อมูลประเภทที่เรียงลำดับ (sequential file) แล้ว
การทำ file maintenance ก็หมายถึงการปรับแก้ข้อมูล ในแฟ้มข้อมูลนั้นให้ตรงกับความจริง
ตามกาลเวลา ที่เราเรียกว่า update รวมถึงการกำหนดชื่อของข้อมูล ตำแหน่งของข้อมูล การ
เพิ่มเติมข้อมูล และการตัดข้อมูลที่ไม่ว่างออก

ในกรณีที่แฟ้มข้อมูลใดที่จัดเก็บอยู่ในหน่วยความจำหลักนั้น กิจกรรมทั้งหลายที่กล่าวมานี้
ก็ยังคงดำเนินการได้โดยง่าย ทั้งนี้เพราะอาศัยกรรมวิธีการเรียงลำดับ การรวมข้อมูล และ
การสืบค้น ส่วนการกำหนดที่ตั้งของข้อมูลหรือที่เรียกว่า posting นั้นจะประกอบด้วยขั้นตอน

การสืบค้นเพื่อหาค่าแห่งคองตามวัตถุประสงค์ที่ต้องการ และเมื่อพบค่าแห่งของข้อมูลที่ต้องการแล้ว จึงทำการเปลี่ยนแปลงรายการข้อมูลในระเบียบข้อมูลนั้นโดยตรง ส่วนการเพิ่มเติมหน่วยระเบียบข้อมูลลงไปเพิ่มข้อมูลนั้น ก็ทำโดยการเพิ่มระเบียบข้อมูลใหม่เข้าไปต่อท้ายค่าแห่งสุดท้ายของข้อมูลในเพิ่มข้อมูลนั้นลงไป ในกรณีที่เพิ่มข้อมูลนั้นเป็นการเก็บข้อมูลชนิดใหม่เรียงลำดับตามเขตข้อมูลหลัก แต่ถ้าเป็นกรณีของเพิ่มข้อมูลชนิดที่เรียงลำดับ การที่จะแทรกระเบียบข้อมูลที่เพิ่มเติมลงไปให้อยู่ในลำดับที่ถูกต้องนั้นจำเป็นจะต้องอาศัยกรรมวิธีของการเรียงลำดับเข้าช่วยซึ่งอาจจะใช้วิธีการของ insertion sort เข้าช่วยเพื่อหาค่าแห่งที่ถูกคองที่จะแทรกระเบียบข้อมูลลงไป ส่วนกรรมวิธีของการลบระเบียบข้อมูลบางอันที่ไม่ต้องการทิ้งไปนั้น ทำได้โดยการค้นหาระเบียบข้อมูลที่ลบทิ้งว่าปรากฏอยู่ในตำแหน่งใดในลิสต์ เมื่อพบแล้วจึงทำการลบระเบียบข้อมูลนั้นทิ้งไป โดยการเคลื่อนย้ายระเบียบข้อมูลระดับล่างลงไปขยับขึ้นมาแทนที่ตำแหน่งของข้อมูลที่เราจะลบทิ้งไปนั้น ในบางครั้งเราอาจจะไม่จำเป็นต้องดำเนินการคองที่กล่าวมาแล้วก็ได้ เพียงแต่กำหนดเขตข้อมูลหลัก ของระเบียบข้อมูลที่เราจะคองทิ้งว่า "deleted" แล้วก็พอ คองนั้นระเบียบข้อมูลนั้นก็ยังคงอยู่แต่เขตข้อมูลหลัก จะเป็น "deleted" เท่านั้นเองวิธีนี้คองง่ายที่ไม่คองไปเคลื่อนย้ายระเบียบข้อมูลลำดับถัดไปให้มาแทนที่ เพราะจะทำให้เสียเวลาถ้ามีข้อมูลในลำดับที่ถัดไปจากตำแหน่งที่จะลบทิ้งมีมาก แต่ก็ยังมีจุดบกพร่องอยู่บ้างเพราะทุกครั้งที่เราทำการค้นหาข้อมูลในลิสต์ เรายังคองเปรียบเทียบ argument กับค่าของเขตข้อมูลหลักที่ deleted แล้ว

การรักษาโครงสร้างของเพิ่มข้อมูลชนิดที่เก็บอยู่ในสื่อต่างๆ ที่เป็นหน่วยความจำสำรอง เช่น เทป จานแม่เหล็กนั้น จำเป็นจะต้องคำนึงถึงคือค่าแห่งที่คองของระเบียบข้อมูลในสื่อซึ่งเรียกว่า physical record หรือ physical block ทั้งนี้เพราะเราไม่สามารถที่จะรับหรือส่งต่อข้อมูลไปยังแต่ละระเบียบข้อมูลได้โดยตรง เหมือนกับลักษณะของข้อมูลที่เก็บในหน่วยความจำหลัก แต่เราคองคำนึงถึง physical record ของระเบียบข้อมูลด้วย ประการที่สอง เราไม่สามารถที่จะเข้าถึงระเบียบข้อมูลตามค่าแห่งที่เก็บอยู่ในเพิ่มข้อมูลได้ทันทีแต่เราคองดำเนินการสร้างเพิ่มข้อมูลอีกครั้งเสียก่อน คองนั้นขั้นตอนต่างๆ ที่เราคองดำเนินการกับเพิ่มข้อมูล เช่น การลบ/การเพิ่มระเบียบข้อมูลและอื่นว จึงแตกต่างจากกรรมวิธีที่ใช้ดำเนินการกับข้อมูลที่จัดเก็บอยู่ในหน่วยความจำหลัก เช่น การดำเนินการค้นหาข้อมูลแบบเรียงลำดับที่ดำเนินการกับ

external file ที่จัดเก็บในหน่วยความจำสำรองนั้นก็มีวิธีที่แตกต่างไปจากที่เคยเกี่ยวข้องกับ internal file ในขั้นตอนที่จะต้องเกี่ยวข้องกับอุปกรณ์ I/O ในการที่จะต้องอ่านที่ละระเบียนจาก external file นั้นเข้ามาเก็บภายในหน่วยความจำหลักก่อนแล้วจึงค่อยดำเนินการค้นหาเปรียบเทียบ ต่อไป

การ posting กับ external file นั้นก็ทำได้ง่ายเช่นเดียวกับการ posting แห่งข้อมูลที่อยู่ในหน่วยความจำหลัก ซึ่งมีวิธีการดังนี้คือข้อมูลในแฟ้มข้อมูลจะถูกอ่านเข้ามาที่ละระเบียนข้อมูลจนกระทั่งพบระเบียนข้อมูลที่เราต้องการ แล้วจึงทำการเปลี่ยนแปลงรายการข้อมูลของระเบียนข้อมูลนั้นภายในที่เก็บซึ่งอยู่ในสมอง เครื่อง หลังจากนั้นจึงนำข้อมูลทั้งหมดภายในระเบียนข้อมูลนั้นบันทึกซ้ำกับไปยังตำแหน่งเดิมใน external file ในบางครั้งนั้นเราอาจจะต้องดำเนินการ posting หลาย ๆ ครั้งในแฟ้มข้อมูลใดแฟ้มข้อมูลหนึ่ง ทั้งนี้การที่จะทำการ posting นั้นมีประสิทธิภาพยิ่งขึ้นก็สามารถทำได้โดยการสร้างเป็น "posting file" ชนิดเรียงลำดับขึ้นมาโดยที่ posting file นี้จะถูกจัดเรียงลำดับเช่นเดียวกับการเรียงลำดับในแฟ้มข้อมูลใหม่ที่ต้องการ Posting file ที่สร้างขึ้นนี้จะช่วยให้กระบวนการ posting ได้รับความสำเร็จเพียงการวนทำงานเพียง 1 รอบเท่านั้นเอง จะได้แฟ้มข้อมูลใหม่ในลักษณะที่ต้องการ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย