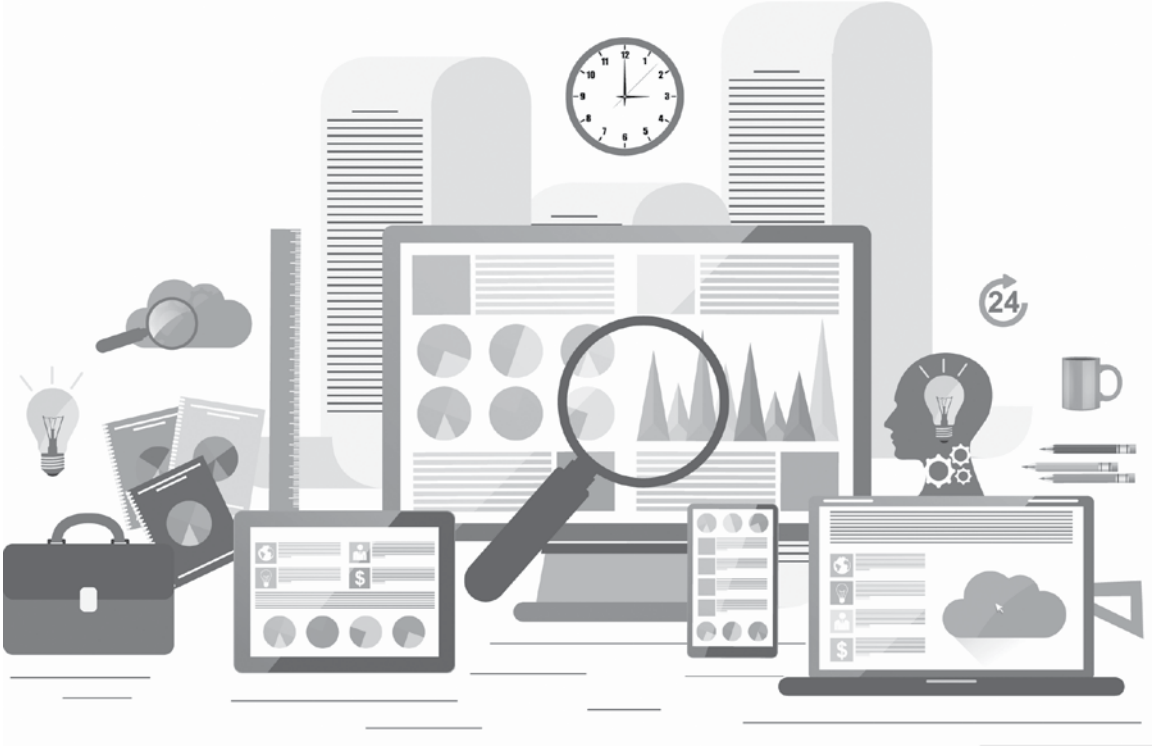

เตรียมรับมือ Big Data Crisis ด้วย Data Management

ต่อ จากฉบับที่แล้ว

วิชญ์คุทธ์ เมาระพงษ์
ที่ปรึกษาโครงการสารสนเทศของหน่วยงานภาครัฐ
สังกัดสถาบันวิจัยและให้คำปรึกษา
แห่งมหาวิทยาลัยธรรมศาสตร์

ถึง แม้ว่าการบริหารจัดการข้อมูลด้วยแฟ้มข้อมูลจะสามารถดำเนินการอย่างเป็นระบบมีความเป็นอิสระในการจัดการโดยบุคลากรหรือหน่วยงานย่อยภายในองค์การที่เป็นเจ้าของข้อมูล ทำให้มีความคล่องตัวค่อนข้างสูง แต่ก็มีข้อด้อยหลายประการดังนี้

■ **เกิดความซ้ำซ้อนของข้อมูล** (data redundancy) เนื่องจากแต่ละบุคคล ฝ่ายหรือหน่วยงานย่อยภายในองค์การมีแฟ้มข้อมูลของตนเอง นั่นคือมีข้อมูลชุดเดียวกันแต่มีการจัดเก็บในแฟ้มข้อมูลที่ต่างกัน หรือข้อมูลชุดเดียวกันถูกจัดเก็บอยู่ในสองแฟ้มข้อมูลหรือมากกว่าซึ่งจะทำให้เป็นการสิ้นเปลืองเนื้อที่ แรงงาน และทรัพยากรที่ใช้ในการจัดเก็บข้อมูลที่ซ้ำซ้อนนั้น

■ **ยากต่อการแก้ไข** (updating difficulties) ความซ้ำซ้อนของข้อมูลจะทำให้ยากต่อการแก้ไขข้อมูลเหล่านั้นเนื่องจากถ้ามีข้อมูลใดเปลี่ยนแปลงจะต้องทำการเปลี่ยนแปลงข้อมูลทุกแฟ้มข้อมูลที่มีข้อมูลซ้ำกันทั้งหมดทำให้อาจเกิดข้อผิดพลาดได้ และเกิดความสับสนหากข้อมูลในแต่ละแฟ้มข้อมูลไม่ตรงกันรวมทั้งสิ้นเปลืองแรงงาน และทรัพยากรในการเปลี่ยนแปลงแก้ไขข้อมูลที่ซ้ำซ้อนนั้นด้วย

■ **เกิดความขัดแย้งของข้อมูล** (data inconsistency) เป็นปัญหาที่มีเกิดจากการจัดเก็บข้อมูลที่ซ้ำซ้อนเนื่องจากการจัดเก็บข้อมูลชุดเดียวกันในหลายแฟ้มข้อมูลอาจทำให้ข้อมูลชุดเดียวกันมีค่าที่แตกต่างกันได้ในแต่ละแฟ้มข้อมูลถ้ามีการแก้ไขปรับปรุงข้อมูลไม่ครบถ้วนซึ่งทำให้ไม่ทราบว่าข้อมูลชุดใดคือข้อมูลที่ถูกต้องที่สุด

จากข้อด้อยดังกล่าวของการจัดการข้อมูลด้วยแฟ้มข้อมูล จึงเป็นที่มาของการพัฒนาระบบการจัดการข้อมูลอีกรูปแบบหนึ่ง เพื่อแก้ปัญหาที่เกิดขึ้นของการจัดการข้อมูลในระบบแฟ้มข้อมูล ซึ่งเรียกว่าระบบการจัดการฐานข้อมูล

ระบบการจัดการฐานข้อมูล

ระบบจัดการฐานข้อมูลนั้น ถ้านิยามตามที่กล่าวถึงข้างต้นก็จะหมายความถึงระบบที่ใช้บริหารจัดการแฟ้มข้อมูล (File) ที่มีความสัมพันธ์กันอย่างมีระเบียบด้วยรูปแบบที่เหมาะสม โดยจะมองแฟ้ม

ข้อมูลในลักษณะของตาราง (Table) ที่ประกอบด้วยระเบียนต่างๆ (Record) ซึ่งระเบียนก็จะประกอบไปด้วยเขตข้อมูล (Field) ที่สอดคล้องกัน ซึ่งการจัดการข้อมูลในลักษณะของฐานข้อมูลทำให้ผู้ใช้งานสามารถใช้ข้อมูลที่เกี่ยวข้องในระบบงานต่างๆ ร่วมกันได้ โดยที่จะไม่เกิดความซ้ำซ้อนของข้อมูล (ระบบฐานข้อมูลจะลดความซ้ำซ้อนของข้อมูลในขั้นตอนการออกแบบโครงสร้างด้วย Normalization) และยังสามารถหลีกเลี่ยงความขัดแย้งของข้อมูลได้ ซึ่งจะส่งผลให้ข้อมูลในระบบถูกต้องเชื่อถือได้ และเป็นมาตรฐานเดียวกัน รวมถึงมีการกำหนดระบบความปลอดภัยของข้อมูลขึ้น โดยเราจะได้เรียกระบบจัดการฐานข้อมูล(เชิงสัมพันธ์) ว่า Relational Database Management System (RDBMS) ที่เพิ่มคำว่า “เชิงสัมพันธ์” ต่อท้ายฐานข้อมูลเนื่องจากระบบฐานข้อมูลที่ใช้กันอยู่ในปัจจุบันซึ่งทำงานร่วมกับระบบสารสนเทศขององค์การเป็นฐานข้อมูลเชิงสัมพันธ์ มีข้อกำหนดในการออกแบบโครงสร้างที่เน้นความเชื่อมโยง และสอดคล้องกันของข้อมูลในลักษณะของ Row Column ซึ่งจะสามารถใช้ได้ดีกับข้อมูลที่เป็นข้อความ และตัวเลข รวมถึงการจัดเก็บไฟล์รูปภาพโดยแปลงเป็นข้อมูลเป็น Binary Digit สามารถเรียกใช้งานโดย Structured Query Language หรือภาษา SQL

แม้ว่าระบบฐานข้อมูลเชิงสัมพันธ์จะมีประสิทธิภาพสูงแต่มีข้อจำกัดในการทำงานร่วมกับข้อมูลในรูปแบบใหม่ๆ ที่เป็น Multimedia, โครงสร้างทางวิศวกรรม, 3D Model ฯลฯ ซึ่งมีความซับซ้อนมากกว่า จึงได้มีการพัฒนาระบบฐานข้อมูลในลักษณะ Object-oriented Database Management System (ODBMS) โดยจะเก็บข้อมูลในรูปแบบ Object ซึ่งเก็บตัว Object, Attribute หรือคุณสมบัติ และ Method ของ Object ไว้ในฐานข้อมูล การใช้งานสามารถทำได้ผ่านโปรแกรมภาษาประเภท Object-oriented เช่น Java หรือ C++ หรือใช้ภาษา SQL ในการติดต่อฐานข้อมูล ตัวอย่างของฐานข้อมูลแบบ ODBMS ได้แก่ Versant, Object store, Jasmine และ Poet

โครงสร้างของ ODBMS นั้นมีความยืดหยุ่นกว่า แบบ RDBMS คือไม่จำเป็นต้องมีโครงสร้างแบบ Row Column และสามารถมีความสัมพันธ์ที่ซับซ้อนระหว่างข้อมูลได้ เป็นการลดขั้นตอน และเวลาในการออกแบบฐานข้อมูลลงไปได้ อย่างไรก็ตาม ODBMS ยังไม่เป็นมาตรฐาน มีการพัฒนาปรับปรุงอยู่อย่างต่อเนื่องโดยผู้ผลิตแต่ละรายในแบบต่างคนต่างทำ ทำให้แต่ละผลิตภัณฑ์ก็มีจุดเด่นแตกต่างกันออกไปซึ่งเหมาะกับการจัดการข้อมูลแบบเฉพาะทาง แต่ ODBMS และ RDBMS สามารถใช้งานร่วมกันได้ และมักจะพบว่าจะถูกใช้งานควบคู่กันไป เพราะทั้งสองถูกออกแบบมาสำหรับงานที่แตกต่างกัน ODBMS ถูกใช้สำหรับงานเฉพาะทางที่มีความซับซ้อน ส่วน RDBMS จะถูกใช้กับข้อมูลทั่วไปประเภท Descriptive Data เช่น การเก็บชื่อ ที่อยู่ เบอร์โทรศัพท์ ตัวเลข

สำหรับกรณีของการจัดการ Big Data ทำให้มีการต่อยอดแนวคิดของ RDBMS และ ODBMS ที่มีการบริหารจัดการอย่างเป็นระบบแต่มีข้อจำกัดที่โครงสร้างความสัมพันธ์ กล่าวคือเมื่อระบบมีขนาดใหญ่ขึ้นโครงสร้างข้อมูลที่เป็นระเบียนหรือตาราง และข้อมูลมีจำนวนมากขึ้น ความสัมพันธ์ระหว่างข้อมูลก็มีมากขึ้นเช่นกัน ทำให้เกิดความซับซ้อนที่ส่งผลกระทบต่อความสามารถ และความเร็วในการจัดเก็บ ค้นหา และประมวลผลข้อมูลอาจจำเป็นต้องใช้ของระบบคอมพิวเตอร์ที่มีศักยภาพสูงซึ่งมีราคาแพงหรือลงทุนในระดับของ Cluster หรือ Cloud รวมถึงข้อจำกัดในการจัดเก็บข้อมูลในรูปแบบอื่นๆ ที่นอกเหนือจากที่ RDBMS จัดการได้ การออกแบบโครงสร้างในการจัดเก็บที่ต้องการลดความซ้ำซ้อนด้วย Normalization อาจไม่ความจำเป็นมากนักเนื่องจากส่วนใหญ่ข้อมูลจะถูกใช้เพื่อประมวลผลในลักษณะของแนวโน้ม และการคาดการณ์ นำไปสู่แนวคิด และการพัฒนา NoSQL Database ขึ้น

NoSQL Database ที่มีการนำมาใช้งานนั้น มีอยู่ด้วยกัน 4 ชนิด ประกอบด้วย

1. Column-Oriented Database RDBMS ที่นิยมใช้งานในปัจจุบันจะเป็น Row-based oriented นั่นคือ แต่ละ Row หรือระเบียนของข้อมูล (Record) จะประกอบไปด้วย Row ID และ field หรือ column ต่างๆ โดยแต่ละ Row จะถูกจัดเก็บในตาราง โดยการดึงข้อมูลจากตารางจะเป็นแบบอ่านข้อมูลบนลงล่าง และซ้ายไปขวา ทำให้เสียเวลา และใช้ทรัพยากรหน่วยความจำ (Memory) เป็นอย่างมาก

ROWID	Name	Birthday	Hobbies
1	Jos The Boss	11-12-1985	archery, conquering the world
2	Fritz von Braun	27-1-1978	building things, surfing
3	Freddy Stark		swordplay, lollygagging, archery
4	Delphine Thewiseone	16-9-1986	

Row-oriented lookup: from top to bottom and for every entry all columns are taken in memory.

ดังนั้น Column-Oriented Database จึงสร้างมาเพื่อช่วยแก้ไขปัญหาเหล่านี้โดยแต่ละ Column จะถูกจัดเก็บแยกกันทำให้การเข้าถึงข้อมูลในแต่ละ column เร็วขึ้น รวมทั้งทำให้ง่ายต่อการบีบอัดข้อมูลอีกด้วยเนื่องจากในแต่ละตารางจัดเก็บข้อมูลเพียงชนิดเดียว

Name	ROWID
Jos The Boss	1
Fritz Schneider	2
Freddy Stark	3
Delphine Thewiseone	4

Birthday	ROWID
11-12-1985	1
27-1-1978	2
16-9-1986	4

Hobbies	ROWID
archery	1, 3
conquering the world	1
building things	2
surfing	2
swordplay	3
lollygagging	3

A column-oriented database stores each column separately

ตัวอย่าง Column-Oriented Database เช่น Apache HBase, Cassandra, Hypertable, Google BigTable เป็นต้น

อ่านต่อฉบับหน้า