

แบบรูป ลักษณะคุณสมบัติและข้อดีของฐานข้อมูลเชิงวัตถุ

Object-Oriented Database Patterns, Characteristics and Advantages

ปณวิทย์ เหมาะประสิทธิ์¹, สุปราณี วงษ์แสงจันทร์²

บทคัดย่อ

ในปัจจุบัน เทคโนโลยีเชิงวัตถุเข้ามามีบทบาทอย่างมากกับทุกส่วนของระบบสารสนเทศ ไม่ว่าจะเป็นการเขียนโปรแกรมเชิงวัตถุ การพัฒนาซอฟต์แวร์เชิงวัตถุ รวมไปถึงฐานข้อมูลเชิงวัตถุอีกด้วย ซึ่งถือเป็นรูปแบบฐานข้อมูลที่แตกต่างกันจากฐานข้อมูลเชิงสัมพันธ์เป็นอย่างมาก โดยคุณสมบัติที่เห็นได้เด่นชัดคือ ฐานข้อมูลเชิงวัตถุจะเก็บข้อมูลต่างๆ ไว้ในออบเจกต์ แต่สำหรับฐานข้อมูลเชิงสัมพันธ์จะเก็บข้อมูลไว้ในตาราง และฐานข้อมูลเชิงวัตถุนั้นยังมีข้อดีที่สำคัญคือ สามารถจัดการข้อมูลที่มีความซ้ำซ้อนได้ ซึ่งในจุดนี้ฐานข้อมูลเชิงสัมพันธ์ไม่สามารถจัดเก็บได้โดยตรง ซึ่งในการศึกษาค้นคว้าครั้งนี้จะกล่าวถึง ความเป็นมา ลักษณะคุณสมบัติและข้อดีของฐานข้อมูลเชิงวัตถุ เมื่อเปรียบเทียบกับฐานข้อมูลเชิงสัมพันธ์ที่เป็นรูปแบบฐานข้อมูลที่เป็นที่นิยมอย่างแพร่หลายในปัจจุบัน

คำสำคัญ: ฐานข้อมูลเชิงวัตถุ ฐานข้อมูลเชิงสัมพันธ์ แบบจำลองฐานข้อมูล เทคโนโลยีเชิงวัตถุ

Abstract

Nowadays, object-oriented technology is used in many areas of information systems including Object-Oriented Analysis, Object-Oriented Design, Object-Oriented Programming, and Object-Oriented Database. Nevertheless, the Object-Oriented Database is totally different from the Relational Database. The major different characteristic is that Object-Oriented Database stores data in an object while Relational Database keeps data in relational tables. In addition, Object-Oriented Database has some brilliant advantages that it can manage complex data which Relational Database cannot handle. This paper presents the history and also the characteristics of Object-Oriented Database. Moreover, we will reveal the advantages of Object-Oriented Database in comparison to Relational Database which is the most selected database in the market.

¹คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

²อาจารย์ประจำ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยอีสเทิร์นเอเซีย

ความนำ

การใช้ฐานข้อมูลถือเป็นตัวเลือกอันดับแรกๆ ของการจัดเก็บข้อมูลของระบบสารสนเทศในปัจจุบัน เนื่องจากสามารถจัดการข้อมูลได้อย่างมีประสิทธิภาพ และสามารถนำข้อมูลที่เก็บไว้ไปประมวลผลได้ดีกว่าแบบแฟ้มข้อมูล ซึ่งเป็นรูปแบบการจัดเก็บข้อมูลในแบบดั้งเดิม อีกทั้งการเก็บข้อมูลแบบดั้งเดิมเพราะอาจเกิดปัญหาขึ้นได้หลายอย่าง อาทิ ความซ้ำซ้อนของข้อมูลเพราะไม่ได้เก็บข้อมูลไว้ที่เดียวกัน และทำให้ข้อมูลไม่เกิดบูรณาภาพ จากปัญหาดังกล่าวข้างต้น ฐานข้อมูลจึงเป็นตัวเลือกที่ดีเพื่อใช้ในการจัดเก็บข้อมูลของระบบสารสนเทศ

อย่างไรก็ตาม ฐานข้อมูลนั้นสามารถแบ่งแยกได้อีกหลายแบบ และที่นิยมใช้อย่างมากในปัจจุบันก็คือ ฐานข้อมูลเชิงสัมพันธ์ เหตุเพราะใช้งานง่าย ไม่ยุ่งยาก ผู้ใช้งานสามารถทำความเข้าใจในการใช้งานได้ไม่ยาก และยังมีซอฟต์แวร์ที่สามารถใช้งานได้ร่วมกันอยู่มากมาย แต่ยังมีฐานข้อมูลอีกรูปแบบหนึ่ง คือ ฐานข้อมูลเชิงวัตถุ ซึ่งในปัจจุบันเทคโนโลยีเชิงวัตถุได้เข้ามามีบทบาทเป็นอย่างมากกับระบบสารสนเทศ จึงทำให้เกิดฐานข้อมูลเชิงวัตถุขึ้นมา โดยฐานข้อมูลเชิงวัตถุมีความแตกต่างและสามารถแก้ไขข้อจำกัดบางอย่างของฐานข้อมูลเชิงสัมพันธ์ได้ จึงเป็นจุดที่น่าพิจารณาว่า ฐานข้อมูลเชิงวัตถุมีข้อดีที่เหนือกว่าฐานข้อมูลเชิงสัมพันธ์ และฐานข้อมูลเชิงวัตถุจะสามารถทดแทนฐานข้อมูลเชิงสัมพันธ์ได้หรือไม่ในอนาคตอันใกล้

ความเป็นมาของฐานข้อมูล

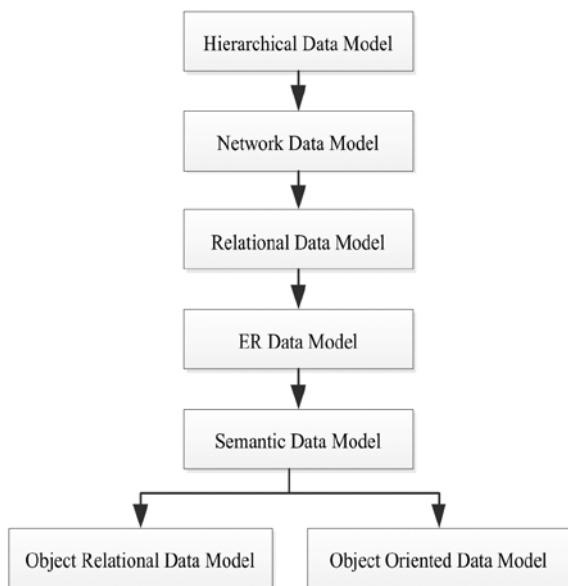
ฐานข้อมูลที่ถูกใช้อยู่จริงในปัจจุบัน ประกอบด้วย ฐานข้อมูลเชิงสัมพันธ์และฐานข้อมูลเชิงวัตถุ

ฐานข้อมูลเชิงสัมพันธ์

ฐานข้อมูลเชิงสัมพันธ์ (Stajano, 1998) นั้นได้ถือกำเนิดขึ้นในช่วงปีคริสต์ศักราชที่ 1969 และได้รับการยอมรับอย่างรวดเร็ว เนื่องจากฐานข้อมูลเชิงสัมพันธ์สามารถทำความเข้าใจได้ง่ายทำให้ได้รับการตอบรับเป็นอย่างดีกับผู้ใช้ในช่วงนั้น ทุกคนเชื่อว่าฐานข้อมูลเชิงสัมพันธ์ที่ใช้หลักการทฤษฎีความสัมพันธ์ของข้อมูลมาจัดการกับข้อมูลนั้นเป็นสิ่งที่ดี อย่างไรก็ตาม ต้องใช้เวลาถึงสิบปีก่อนที่ระบบเชิงพาณิชย์จะสอดคล้องกับทฤษฎีความสัมพันธ์

ฐานข้อมูลเชิงวัตถุ

สำหรับฐานข้อมูลเชิงวัตถุ (Kim, 1990) นั้นมีจุดกำเนิดในช่วงกลางปีคริสต์ศักราชที่ 1980 เนื่องจากการเขียนโปรแกรมเชิงวัตถุได้รับความนิยมในการใช้งานเป็นอย่างมาก จึงทำให้มีความต้องการในการใช้ระบบฐานข้อมูลเชิงวัตถุมากขึ้นตามไปด้วย ประกอบกับการค้นพบขีดจำกัดของฐานข้อมูลเชิงสัมพันธ์ ตัวอย่างเช่น การระบุความสัมพันธ์ของข้อมูลระหว่างรีเลชั่นที่ไม่ชัดเจน การเก็บข้อมูลไม่สื่อถึงสิ่งที่มีอยู่จริง จะต้องไม่มีชุดรายการ หรือการยอมให้ตารางอยู่ภายในสคีม์ รวมทั้งฐานข้อมูลเชิงสัมพันธ์นั้นไม่เหมาะสมกับโครงสร้างข้อมูลที่มีความซับซ้อน หรือประเภทข้อมูลใหม่ที่มีขนาดใหญ่และวัตถุที่ไม่มีโครงสร้าง ด้วยปัจจัยดังกล่าวจึงทำให้ฐานข้อมูลเชิงวัตถุ เริ่มเข้ามามีบทบาทมากขึ้นและขยายไปสู่ธุรกิจฐานข้อมูล ในปลายปีคริสต์ศักราช 1980 วิวัฒนาการของแบบจำลองข้อมูล (T.M. and C.E., 2002) แสดงดังรูป 1



ภาพ 1 วิวัฒนาการของแบบจำลองข้อมูล

รายละเอียดของฐานข้อมูลเชิงวัตถุ

แนวคิด ลักษณะ และคุณสมบัติ

ฐานข้อมูลเชิงวัตถุ (มหาวิทยาลัยราชพฤกษ์, 2552) เกิดจากแนวคิดของการโปรแกรมเชิงวัตถุ (Object Oriented Program--OOP) ที่พิจารณาของทุกสิ่งเป็นวัตถุ โดยในแต่ละวัตถุเป็นแหล่งรวมของข้อมูลและการปฏิบัติงานที่มีคลาสเป็นตัวกำหนดคุณสมบัติหรือรายละเอียดของวัตถุรวมทั้งคุณสมบัติ การปกปิดความลับของวัตถุ สำหรับการเข้าถึงข้อมูลนั้นต้องมีการตอบรับจากเมธอด (method) ในวัตถุนั้นว่าจะอนุญาตในการส่งข้อความ (message) เพื่อการติดต่อหรือไม่ โดยหลักการของฐานข้อมูลเชิงวัตถุ ควรมีลักษณะและคุณสมบัติ (Kratomza, 2008) ดังต่อไปนี้

1. คลาส (Class) คือต้นแบบข้อมูลที่มีไว้เพื่อสร้างอ็อบเจกต์ โดยอ็อบเจกต์ทุกตัวจะต้องอยู่ในคลาสเสมอ หากต้องการคุณสมบัตินั้น (characteristic) ของอ็อบเจกต์สามารถดูได้จากคลาส ซึ่งคลาสจะมีลักษณะเป็นกลุ่มของอ็อบเจกต์ที่มีโครงสร้างพื้นฐานพฤติกรรมเดียวกัน ดังนั้นอ็อบเจกต์ที่มีคุณสมบัตินั้นลักษณะเดียวกันนี้ ก็จะรวมกลุ่มอยู่ในคลาสเดียวกัน

2. อ็อบเจกต์ (Object) คือหน่วยที่ระบบสนใจ

ซึ่งทำให้เกิดเหตุการณ์บางอย่าง อ็อบเจกต์จะเน้นความสัมพันธ์ระหว่างอ็อบเจกต์ในระบบเป็นหลัก สำหรับการกระทำใด ๆ ก็ตาม จะถือว่าเป็นการปฏิบัติที่เกิดขึ้นจากอ็อบเจกต์หนึ่งไปมีความสัมพันธ์กับอีกอ็อบเจกต์หนึ่งในการสื่อสารกับอ็อบเจกต์อื่น ๆ ของระบบ เพื่อตอบสนองความต้องการของตน เรียกว่า “ร้องขอและตอบสนอง” ซึ่งการขอความช่วยเหลือของอ็อบเจกต์หนึ่งกับอีกอ็อบเจกต์หนึ่งจะเรียกว่า “Message” และที่สำคัญอ็อบเจกต์แต่ละตัวจะมีบทบาทและหน้าที่ของตนเอง หมายถึง อ็อบเจกต์เป็นแหล่งรวมของข้อมูลและโอเปอเรชันในตัวเอง โดยแต่ละอ็อบเจกต์จะประกอบด้วย 2 ส่วน คือ คุณสมบัติ (attribute) ซึ่งจัดเป็นข้อมูลของตนเอง และพฤติกรรม (behavior/method) แสดงการทำงานของตนเองหรือสามารถติดต่อกับอ็อบเจกต์อื่น ๆ ได้ โดยผ่าน Message

3. การสืบทอดคุณสมบัติ (Inheritance) สำหรับการสืบทอดคุณสมบัตินั้นจัดว่าเป็นเรื่องที่สำคัญอย่างหนึ่งของเทคโนโลยีเชิงวัตถุ หลักการของเทคโนโลยีเชิงวัตถุจะขาดคุณสมบัติของการสืบทอดนี้ไม่ได้ เนื่องจากการสืบทอดคุณสมบัตินี้จะทำให้เกิดข้อดีต่างๆ ตามหลักการของเทคโนโลยีเชิงวัตถุ ดังต่อไปนี้

(1) ทำให้โครงสร้างเป็นระบบระเบียบและสามารถปรับเปลี่ยนได้ง่าย

(2) ลดเวลาและค่าใช้จ่ายในการพัฒนาระบบ หลักของการสืบทอดคุณสมบัตินี้ที่เป็นลำดับขั้นนั้นทำให้สามารถแสดงความสัมพันธ์ระหว่างอ็อบเจกต์ที่ชัดเจนยิ่งขึ้น กล่าวคือ หากระบบมีความสัมพันธ์ที่ชัดเจนมากขึ้นเท่าใด ก็จะส่งผลทำให้การออกแบบระบบงานนั้นง่ายขึ้นเท่านั้น ผู้ออกแบบและพัฒนาระบบงานเชิงวัตถุก็จะสามารถออกแบบระบบงานขนาดใหญ่ได้ โดยอาศัยอ็อบเจกต์ที่ได้มีการนิยามไว้แล้ว

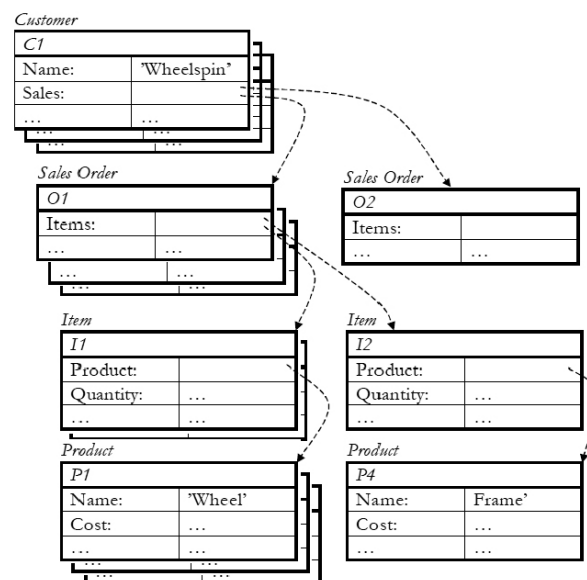
4. โพลิมอร์ฟิซึม (Polymorphism) หมายถึงการออกแบบเดียว แต่สามารถรับการตอบสนองได้หลายรูปแบบ คือ สามารถมีเมธอด (Method) ชื่อเดียวกันได้ในหลายๆ คลาส โดยไม่จำเป็นต้องมีพฤติกรรมหรือการทำงานที่เหมือนกัน ตัวอย่างเช่น การจ่ายค่าแรงพนักงาน

สามารถมีหลายประเภท ได้แก่ พนักงานประจำ พนักงานรายชั่วโมง หรือพนักงานรายสัปดาห์ ซึ่งเมื่อต้องการจ่ายค่าแรงจะมีเพียงตัวเดียวเท่านั้น แต่สามารถแยกการจ่ายค่าแรงตามประเภทต่างๆ ที่ต้องการได้

5. เอ็นแคปซูลชัน (Encapsulation) คือการปกปิดความลับของอ็อบเจกต์ ซึ่งเป็นการรวมคุณสมบัติของอ็อบเจกต์และพฤติกรรม (behavior/method) เข้าไว้ด้วยกัน จะทำหน้าที่ป้องกันมิให้อ็อบเจกต์อื่นที่อยู่ภายนอกเข้าถึงอ็อบเจกต์หนึ่งๆ ได้อย่างอิสระ โดยจะอนุญาตให้เฉพาะพฤติกรรม (behavior/method) ที่อยู่ภายในอ็อบเจกต์เท่านั้นที่สามารถติดต่อกับคุณสมบัติ (attribute) ที่อยู่ภายในอ็อบเจกต์เดียวกันได้

6. Object Identity (OID) เป็นสิ่งที่ระบุถึงความแตกต่างของอ็อบเจกต์แต่ละตัว โดย Object Identity (OID) จะถูกกำหนดให้อ็อบเจกต์ กี่ต่อเมื่อมีการสร้างอ็อบเจกต์ขึ้นมา เปรียบเสมือนเป็นการอนุญาตให้อ็อบเจกต์ในระบบฐานข้อมูลนั้นมีความเป็นอิสระต่อกัน โดยอ็อบเจกต์จะยังคงรักษาความเป็นลักษณะเฉพาะตนอยู่ถึงแม้ว่าข้อมูลของตัวแปรหรือวิธีการของอ็อบเจกต์ จะมีการเปลี่ยนแปลงไป

7. ภาษานิยามวัตถุ (Object Definition Language) ในระบบฐานข้อมูลเชิงสัมพันธ์นั้นจะมีภาษาที่ใช้ในการนิยามข้อมูล (Data Definition Language--DDL) ซึ่งในระบบฐานข้อมูลเชิงวัตถุก็มีภาษาที่ใช้นิยามวัตถุ (Object Definition Language: ODL) เช่นเดียวกัน โดยภาษาที่ใช้นิยามวัตถุนี้จะอนุญาตให้สามารถกำหนดลอจิกคัลสกีมาในฐานข้อมูลเชิงวัตถุ ซึ่งภาษาที่ใช้นิยามวัตถุนี้เป็นภาษาโปรแกรมที่ไม่ขึ้นกับภาษาใดๆ และมักนำไปใช้งานสำหรับในการนิยาม OODB schema โดยภาษานิยามวัตถุ (ODL) นี้สามารถเคลื่อนย้าย (Portable) ไปยัง ODMG-Compliant Data Manipulation Language ซึ่งเป็นภาษาเฉพาะที่ใช้กับฐานข้อมูลเชิงวัตถุ (Bergholt, 1998) ได้ดังภาพ 2



ภาพ 2 ตัวอย่างโมเดลฐานข้อมูลเชิงวัตถุ

ส่วนประกอบ

สำหรับฐานข้อมูลเชิงวัตถุนั้นมองเอนติตีในลักษณะของอ็อบเจกต์ (สมจิตร อาจอินทร์และงามนิจอจอินทร์, 2550) ที่มีองค์ประกอบ 3 ส่วนอยู่ภายใน คือ (1) ชื่ออ็อบเจกต์ (2) คุณสมบัติ (properties) ที่ใช้อธิบายอ็อบเจกต์ ซึ่งเปรียบเหมือนกับแอตทริบิวต์ที่ใช้อธิบายเอนติตี และ (3) เมธอด (Method) หรือฟังก์ชัน (Function) ที่แสดงถึงพฤติกรรมที่สามารถกระทำต่อคุณสมบัติของอ็อบเจกต์นั้นๆ กลุ่มของอ็อบเจกต์ที่มีลักษณะที่เหมือนกันจะถูกจัดอยู่ในกลุ่มเดียวกันเรียกว่า คลาส (Class) ดังนั้นคลาสจึงเป็นที่รวมของอ็อบเจกต์ที่มีคุณสมบัติและเมธอดหรือพฤติกรรมร่วมกัน ตัวอย่างคลาสของอ็อบเจกต์ (Sommerville, 2004) เช่น คลาส “พนักงาน” ประกอบด้วยอ็อบเจกต์ “พนักงาน” ซึ่งมีคุณสมบัติที่ใช้อธิบายอ็อบเจกต์คือ ชื่อพนักงาน ที่อยู่ของพนักงาน รหัสพนักงาน หมายเลขบัตรประจำตัวประชาชน เป็นต้น คุณสมบัติของคลาสนี้ยังใช้ในการสร้างความสัมพันธ์กับคลาสอื่นๆ ได้อีกด้วย นอกจากนี้แล้วคลาส “พนักงาน” ยังประกอบด้วยเมธอดต่างๆ เช่น เมธอดการเข้าทำงานของพนักงาน เมธอดการลาออกของพนักงาน เมธอดการเปลี่ยนแปลงรายละเอียดของพนักงาน กล่าวอีกนัยหนึ่ง เมธอดจึงเปรียบเสมือนกับเป็นฟังก์ชันหรือโพรซีเจอร์ (Procedure)

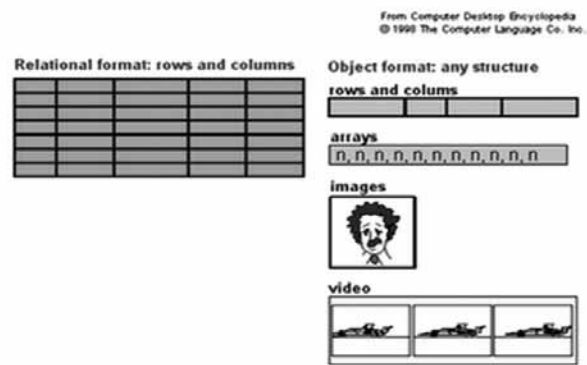
ที่ประกอบด้วยโค้ด (code) ย่อยของภาษาโปรแกรม ที่นักเขียนโปรแกรมสามารถสร้างขึ้นมาเพื่อการเข้าถึงคุณสมบัติที่อยู่ภายในคลาส และการแสดงพฤติกรรมของคลาส ดังภาพ 3



ภาพ 3 ตัวอย่างอ็อบเจกต์พนักงาน

ฐานข้อมูลเชิงวัตถุ	ฐานข้อมูลเชิงสัมพันธ์
อ็อบเจกต์ (object)	เอนติตี้ ทัปเพิล (entity, tuple)
ตัวแปรกรณีตัวอย่าง (instance variable)	สดมภ์ แถว (column, row)
ลำดับชั้นของคลาส (class hierarchy)	แผนฐานข้อมูล แบบ (database scheme)
คลาสคอลเลกชัน (collection class)	รีเลชัน (relation)
อ็อบเจกต์ ไอดีเฟนติไฟเออร์ (object identifier : OID)	คีย์ (key)
การส่งข้อความ (message)	การเรียกโพรซีเจอร์ (procedure call)
เมธอด (method)	ส่วนโพรซีเจอร์ (procedure body)

ภาพ 4 เปรียบเทียบความแตกต่างของส่วนที่ทำหน้าที่คล้ายกันของฐานข้อมูลเชิงวัตถุกับฐานข้อมูลเชิงสัมพันธ์



ภาพ 5 ตัวอย่างการเก็บข้อมูลระหว่าง RDBMS และ OODBMS

การเปรียบเทียบกับฐานข้อมูลเชิงสัมพันธ์

ความแตกต่าง

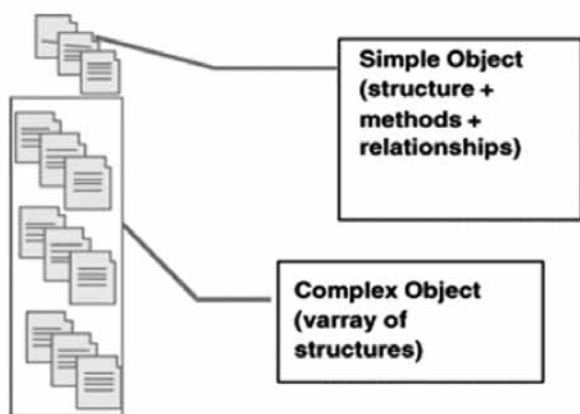
เมื่อเปรียบเทียบส่วนที่ทำหน้าที่คล้ายกันของฐานข้อมูลเชิงวัตถุกับฐานข้อมูลเชิงสัมพันธ์ จะทำให้เห็นความแตกต่างของทั้งสองฐานข้อมูล (Maier, 2001) ตามรูป 4 เป็นการเปรียบเทียบความแตกต่างของส่วนที่ทำหน้าที่คล้ายกันของฐานข้อมูลเชิงวัตถุกับฐานข้อมูลเชิงสัมพันธ์

ข้อดีเมื่อเทียบกับฐานข้อมูลเชิงสัมพันธ์

หากเปรียบเทียบฐานข้อมูลเชิงวัตถุกับฐานข้อมูลเชิงสัมพันธ์แล้วนั้นมีจุดประสงค์ในการเก็บข้อมูลต่างกัน อีกทั้งยังมีข้อดีและข้อเสียแตกต่างกันไป (กระทรวงเทคโนโลยีสารสนเทศ, 2555) ตามรายละเอียดต่อไปนี้

1. มีความง่ายในการจัดการแบบจำลองและข้อมูลที่มีประเภทข้อมูลซับซ้อน (complex data) ผ่านระบบการจัดการฐานข้อมูลเชิงวัตถุ (Object-Oriented Database Management System--OODBMS) โดยไม่ต้องยุ่งยากในการสร้างโปรแกรมประยุกต์เพื่อการจัดการโดยเฉพาะ

โดยประเภทข้อมูลที่ซับซ้อน (Alarm and Wasan, 2006) ในรูป 6 นั้นจะไม่มีเอกลักษณ์หรือไม่สามารถแยกแยะได้โดยธรรมชาติ และมีความสัมพันธ์แบบกลุ่มต่อกลุ่ม (M:N) อยู่เป็นจำนวนมาก



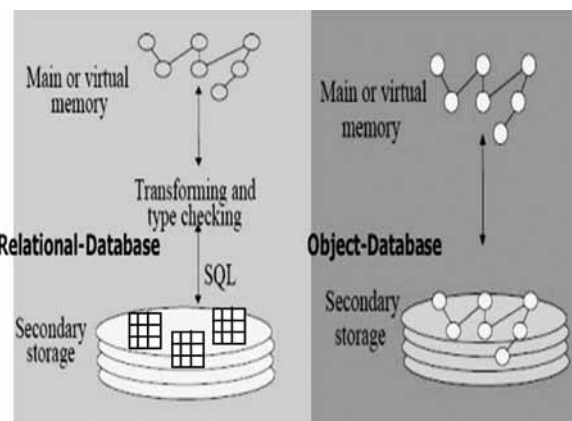
ภาพ 6 รูปแบบข้อมูลที่ซับซ้อนกับข้อมูลทั่วไป

2. จากคุณลักษณะที่สำคัญของฐานข้อมูลเชิงวัตถุ คือ การสืบทอด (Inheritance) รวมถึง Object Normalization จะช่วยให้ลดการเก็บข้อมูลที่มีความซ้ำซ้อนกันในฐานข้อมูล ซึ่งจะส่งผลให้ฐานข้อมูลมีความคงสภาพ (Database Integrity) และลดความขัดแย้งของข้อมูล (Data Inconsistency) ที่จะเกิดขึ้นได้

3. มีความเป็นอิสระทั้งทางด้านข้อมูลและโครงสร้าง (Structural and data independence) ซึ่งจะช่วยให้เกิดความยืดหยุ่นในการขยายฐานข้อมูล ทำให้สามารถปรับปรุงแก้ไข และเข้าถึงข้อมูลได้ง่ายขึ้น

4. ใช้เวลาในการพัฒนาและความยาวของบรรทัดในการเขียนโปรแกรมคอมพิวเตอร์น้อยกว่า เพราะจากการมีคุณสมบัติการสืบทอด (Inheritance) ทำให้มีการใช้ซ้ำ (re-use) ซึ่งเป็นคุณสมบัติที่โดดเด่นอย่างหนึ่งของเทคโนโลยีเชิงวัตถุ

5. การไร้ซึ่งความผิดพลาดในการทำงานร่วมกัน (Lack of impedance mismatch) ในการใช้ฐานข้อมูลเชิงวัตถุนั้น การจะนำข้อมูลมาจัดเก็บจากแอปพลิเคชันที่พัฒนาโดยการโปรแกรมเชิงวัตถุ (Object Oriented Program: OOP) เข้าสู่ฐานข้อมูลนั้น ไม่สามารถทำได้ทันที จะต้องมีการแมปปิง (mapping) เพื่อทำให้สามารถนำมาจัดเก็บในตารางของฐานข้อมูลเชิงสัมพันธ์ก่อนเสมอ ทำให้สูญเสียประสิทธิภาพ (Wade, 2005) ของระบบ โดยการสูญเสียประสิทธิภาพนี้เรียกว่า ความผิดพลาดในการทำงานร่วมกัน (Impedance mismatch) ในภาพ 7



ภาพ 7 ลักษณะของความผิดพลาดในการทำงานร่วมกัน

แต่สำหรับฐานข้อมูลเชิงวัตถุนั้น สามารถนำข้อมูลมาจัดเก็บจากแอปพลิเคชันที่พัฒนาโดยการโปรแกรมเชิงวัตถุ (Object Oriented Program--OOP) เข้าสู่ฐานข้อมูลได้โดยตรง ทำให้ไม่ต้องสูญเสียประสิทธิภาพของระบบ

ตัวอย่างของแอปพลิเคชันที่ใช้ร่วมกับฐานข้อมูลเชิงวัตถุ มีดังต่อไปนี้

- การใช้คอมพิวเตอร์ช่วยในการออกแบบ (Computer-Aided Design--CAD)
- การใช้คอมพิวเตอร์ช่วยในการผลิต (Computer-Aided Manufacturing--CAM)
- การใช้ซอฟต์แวร์ในการออกแบบและพัฒนาซอฟต์แวร์ตัวอื่นๆ (Computer-Aided Software Engineering--CASE)
- ระบบการบริหารจัดการเครือข่าย (Network Management Systems)
- ระบบสารสนเทศในสำนักงาน และระบบมัลติมีเดีย (Office Information Systems--OIS and Multimedia Systems)
- สำนักพิมพ์ดิจิทัล (Digital Publishing)
- ระบบสารสนเทศภูมิศาสตร์ (Geographic Information Systems--GIS)
- เว็บไซต์เชิงโต้ตอบและเว็บไซต์ที่มีการเปลี่ยนแปลงอยู่ตลอดเวลา (Interactive and Dynamic Websites)

- แอปพลิเคชันต่างๆ ที่มีความซับซ้อน รวมถึงข้อมูลที่มีความสัมพันธ์ระหว่างอ็อบเจกต์และโพรซีเยอร์

โดย OODBMS จะเก็บข้อมูลในรูปอ็อบเจกต์ ซึ่งเก็บตัวอ็อบเจกต์ คุณสมบัตินี้ (attribute) และเมธอด (method) ของอ็อบเจกต์ไว้ในฐานข้อมูล สำหรับการใช้งานสามารถทำได้ผ่านโปรแกรมที่เขียนตามหลักของ Object-oriented เช่น ภาษา Java หรือ C++ หรืออาจใช้ภาษา SQL ในการติดต่อฐานข้อมูล ตัวอย่างของฐานข้อมูลแบบ OODBMS ได้แก่ Versant, Object store, Jasmine และ Poet เป็นต้น

บทสรุป

จากการศึกษาครั้งนี้แม้จะพบว่าฐานข้อมูลเชิงวัตถุมีข้อดีในเรื่องของการจัดการกับประเภทข้อมูลที่มีความซับซ้อนสูง การไร้ซึ่งความผิดพลาดในการทำงานร่วมกัน (Lack of impedance mismatch) รวมทั้งหลักการถ่ายทอดคุณสมบัติความสัมพันธ์ และข้อดีด้านอื่นๆ แต่ก็เชื่อว่าจะนำมาทดแทนฐานข้อมูลเชิงสัมพันธ์ได้

Relational Database Management System (RDBMS) นั้นถูกออกแบบมาเพื่อเก็บข้อมูลที่เป็นอักขรตัวเลข ถึงแม้ว่าจะสนับสนุนการเก็บข้อมูลที่เป็น Binary ขนาดใหญ่ (Large Objects--LOB) การใช้ฐานข้อมูลแบบ RDBMS ก็ยังไม่มี การสนับสนุนการเก็บข้อมูลประเภทมัลติมีเดีย เช่น รูปภาพ ออดิโอ และวิดีโอ เพราะในการออกแบบนั้น ถูกออกแบบมาเพื่อให้เก็บข้อมูลที่เป็นตัวอักษรเป็นหลัก และข้อมูลจะต้องถูกเก็บในรูปแบบของตารางเท่านั้น ซึ่งเป็นข้อกำหนด RDBMS ที่สามารถใช้เก็บข้อมูล เช่น รูป วิดีโอ และเสียงได้ แต่ไม่สามารถจัดการข้อมูลเหล่านั้นได้โดยตรง ซึ่งต้องใช้โปรแกรมภายนอกในการจัดการ เช่น การแก้ไขรูป การแสดงรูปและเสียง

การเปรียบเทียบและค้นหาทำโดยโปรแกรมเสริมอื่นๆ

ส่วนข้อเสียของระบบฐานข้อมูลเชิงวัตถุก็มีเช่นกัน คือ ยังขาดภาษาสืบค้นข้อมูลที่เป็นมาตรฐาน รวมทั้งยังมีการใช้งานที่ยุ่งยากและซับซ้อน โดยฐานข้อมูลเชิงวัตถุที่น่าจะเหมาะสมกับระบบงานบางระบบเท่านั้น เช่น งานด้านวิศวกรรม งานทางด้านการวิจัย แต่สำหรับงานทางด้านการสนเทศที่มีการประมวลผลข้อมูลทั่วไปนั้น จะเหมาะสมกับฐานข้อมูลเชิงสัมพันธ์มากกว่า และยังมี การพัฒนา Object-Relational Database System ซึ่งสามารถรองรับคุณลักษณะเด่นของฐานข้อมูลเชิงวัตถุและฐานข้อมูลเชิงสัมพันธ์ ทำให้สามารถเลือกใช้ฐานข้อมูลได้เหมาะสมกับระบบงานมากยิ่งขึ้น

อย่างไรก็ตาม ODBMS และ RDBMS ก็ยังสามารถใช้งานร่วมกันได้ และมักจะพบว่าจะถูกใช้งานควบคู่กันไป เพราะทั้งสองถูกออกแบบมาสำหรับงานที่แตกต่างกัน ODBMS ถูกนำมาใช้สำหรับงานที่เกี่ยวข้องกับระบบงานทางด้านเทคโนโลยีสารสนเทศ (Information Technology) ซึ่งต้องการประสิทธิภาพสูงและข้อมูลมีความซับซ้อน เช่น ภาพวาดที่เป็น 3 มิติ แอนิเมชัน หรือข้อมูลอื่นๆ ที่มีความซับซ้อน ส่วน RDBMS จะถูกใช้กับข้อมูลประเภทที่ต้องการคำอธิบายเป็นตัวอักษร เช่น การเก็บที่อยู่ เบอร์โทรศัพท์ รายละเอียดต่างๆ ของข้อมูล

ดังนั้น การเลือกใช้ระบบจัดการฐานข้อมูล (Database Management System--DBMS) จึงขึ้นอยู่กับชนิดของข้อมูลที่ต้องการจัดเก็บและวัตถุประสงค์ของผู้ใช้งาน ซึ่งหากในอนาคตมีความต้องการในการจัดการกับประเภทของข้อมูลที่มีความซับซ้อนมากขึ้น รวมถึง ถ้าฐานข้อมูลเชิงวัตถุมีภาษาสืบค้นข้อมูลที่เป็นมาตรฐานและใช้งานได้ง่ายขึ้น ฐานข้อมูลเชิงวัตถุอาจจะได้รับความนิยมมากขึ้น และมีส่วนแบ่งการตลาดที่มากขึ้นได้อย่างแน่นอน

เอกสารอ้างอิง

- กระทรวงเทคโนโลยีสารสนเทศและการสื่อสาร. (2555). *Object-oriented Database Management System*. ค้นจาก <http://oodbms-ict.blogspot.com/2012>
- มหาวิทยาลัยราชพฤกษ์, สาขาคอมพิวเตอร์ธุรกิจ. (2552). *แนวคิดระบบฐานข้อมูลเชิงวัตถุ*. ค้นจาก http://demo4.rc.ac.th/chap12_p1/chap12_p1_1.html
- สมจิตร อาจินทร์ และงามนิช อาจินทร์. (2550). *หลักการวิเคราะห์และออกแบบระบบฐานข้อมูล* (พิมพ์ครั้งที่ 2). ขอนแก่น: ขอนแก่นการพิมพ์.
- Andrew E. W. (2005). *Hitting The Relational Wall*. Retrieved from <http://www.odbms.org/download/029.01%20Wade%20Hitting%20the%20Relational%20Wall%202005.PDF>
- Connolly, T. M., & Begg, C. E. (2002). *Database systems: A practical approach to design, implementation and management*, Third Edition, Addison-Wesley.
- David, M. (2001). *Object-Oriented Database Theory*. Retrieved from <http://www.csd.uoc.gr/~hy562/Papers/OODBMS.pdf>
- Frank, S. (1998). *A Gentle Introduction to Relational and Object Oriented Databases*. Retrieved from <http://www.cl.cam.ac.uk/~fms27/db/tr-98-2.pdf>
- Kratomza. (2551). *ระบบฐานข้อมูลเชิงวัตถุ (Object-Oriented Database)*. Retrieved from <http://kratomza.wordpress.com/2011/02/28/object-oriented-database/>
- Lan, S. (2004). *Software Engineering*. Seventh Edition, Pearson Education.
- Lisbeth, B. (1998). *Database Management Systems:Relational, Object-Relational,and Object-Oriented Data Models*. Retrieved from <http://www.cit.dk/cot/reports/reports/Case4/05-v1.1/cot-4-05-1.1.pdf>
- Mansaf, A. & Siri, K.W. (2006). *Migration from Relational Database into Object Oriented Database*. Retrieved from <http://www.scipub.org/fulltext/jcs/jcs210781-784.pdf>
- Won, K. (1990). *Research Directions in Object-Oriented Database Systems*. Retrieved from http://cimic.rutgers.edu/~adam/mmms03/MMIS/db_oodb_res_directions.pdf