หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

# บทที่ 2

# แนวคิดและทฤษฎีที่เกี่ยวข้อง

## 2.1 แนวคิดของ หน่วยรายการ [1]

### 2.1.1 หน่วยรายการ (Transaction)

หมายถึง กลุ่มของคำสั่งงาน หรือชุดคำสั่งงานที่ถูกสั่งให้กระทำการ (Execute) กับข้อมูลของฐานข้อมูลโดย การอ่านข้อมูล การเขียนข้อมูล การลบข้อมูล หรือการปรับปรุงข้อมูล ถ้าแต่ละขั้นตอนการทำงานทั้งหมดสำเร็จ แสดงว่า รายการได้ถูกกระทำเสร็จอย่างสมบูรณ์ (commit) แต่ถ้ามีขั้นตอนใดขั้นตอนหนึ่งทำงานไม่สำเร็จ และต้องการค่าข้อมูลเดิมกลับมา แสดงว่า สภาพของข้อมูลต้องถูกเปลี่ยนให้เป็นค่าเดิม (rollback)

### 2.1.2 หน่วยรายการแบบไกล (remote unit of work/remote transaction)

หมายถึง วิธีที่ผู้ใช้ทำรายการด้วยกลุ่มของคำสั่งงาน หรือชุดคำสั่ง เพื่อการสอบถามข้อมูล สำหรับทำการปรับปรุงรายการ เพิ่มรายการ หรือลบรายการ กับระบบฐานข้อมูลที่อยู่ห่างไกลเพียงหนึ่งฐานข้อมูลเท่านั้น กรณีที่ทำรายการไม่สำเร็จ จะต้องมีการตั้งค่าข้อมูลที่ถูกปรับปรุงไปแล้ว ให้มีสภาพเหมือนก่อนการปรับปรุง ลักษณะนี้เป็นการทำรายการหนึ่งรายการ กับฐานข้อมูลเพียงหนึ่งฐานข้อมูล

### 2.1.3 หน่วยรายการแบบกระจาย (distributed unit of work/distributed transaction)

หมายถึง วิธีการที่ผู้ใช้ทำรายการ ด้วยหลายๆชุดคำสั่งงานในแต่ละรายการ เพื่อการสอบถามข้อมูล ปรับปรุงรายการ เพิ่มรายการ หรือลบรายการ กับระบบฐานข้อมูลที่อยู่ห่างไกลมากกว่าหนึ่งฐานข้อมูล โดยในแต่ละหนึ่งชุดคำสั่งงานลจะมีผลกับระบบฐานข้อมูลเพียงหนึ่งฐานข้อมูล กรณีที่ทำรายการไม่สำเร็จ จะต้องมีการตั้งค่าข้อมูลที่ถูกปรับปรุงไปแล้ว ให้มีสภาพเหมือนก่อนการปรับปรุง ลักษณะนี้หนึ่งรายการจะกระทำกับข้อมูลหลายๆฐานข้อมูล ในบางครั้งจะต้องใช้การยืนยันรายการแบบสองขั้นตอน (two-phase commit) ซึ่งเป็นรูปแบบการยืนยันรายการ ที่เพิ่มความถูกต้องของข้อมูล เกิดขึ้นเมื่อมีการยืนยันรายการมากกว่าหนึ่งระบบฐานข้อมูล ในกรณีที่ระหว่างทำรายการมีเหตุการณ์ใดเกิดขึ้น ระบบจะยอมให้มีการทำรายการต่อไปเพื่อยืนยันรายการตามปกติ หรือ จะทำการยกเลิกการยืนยันข้อมูลทั้งหมด (abort transaction) [5]

## 2.2 การจัดรูปแบบระบบแบบกระจาย (Distributed Systems)

การจัดรูปแบบระบบแบบกระจายเป็นระบบที่ประกอบด้วยคุณสมบัติ ดังนี้ [1]

- เป็นระบบที่ระยะทางระหว่างแหล่งข้อมูล จะต้องมีระยะทางห่างไกลกัน (Remote component)
- เป็นระบบที่สามารถทำการติดต่อพร้อมกันระหว่างจุดต่างๆ (Concurrency)
- เป็นระบบที่กรณีมีปัญหาเกิดขึ้น กับบางจุดของแหล่งข้อมูล จุดอื่นๆในระบบจะต้องไม่ได้รับผลกระทบไปด้วย (Isolation of failures)
- เป็นระบบที่เกิดจากการทำงานร่วมกันได้ ของเทคโนโลยีหลายๆ รูปแบบที่แตกต่างกัน (Heterogeneous technology mix)
- เป็นระบบที่ทรัพยากรในระบบที่ทำงานร่วมกัน เช่น อุปกรณ์ต่างๆ และผู้ใช้สามารถที่จะสับเปลี่ยนระหว่างแหล่งข้อมูลได้ (Mobility of component)
- เป็นระบบที่แหล่งข้อมูลแต่ละแห่งมีความเป็นอิสระของตนเอง ไม่สามารถกำหนดสภาพเพื่อครอบคลุมทั้งหมดได้ (Lack of global state)

สาเหตุที่จะต้องมีการจัดระบบแบบกระจาย มีดังนี้ [7]

- เพื่อให้มีการแบ่งทรัพยากรในระบบมาใช้ร่วมกัน เช่น การใช้แฟ้มข้อมูลร่วมกัน การใช้เครื่องพิมพ์ หรือ ใช้อุปกรณ์ฮาร์ดแวร์ อื่นๆ ร่วมกัน
- เพื่อเพิ่มความรวดเร็วในการประมวลผล เช่น การส่งงานที่ต้องใช้การประมวลผลเร็วๆ ไปทำงานบนเครื่องที่อยู่อีกจุดหนึ่ง เรียกว่า การแบ่งภาระงาน
- เพื่อเพิ่มความน่าเชื่อถือ กรณีเครื่องคอมพิวเตอร์ที่จุดหนึ่งไม่สามารถทำงานได้ที่จุดอื่นๆ ยังคงสามารถทำงานได้อย่างปกติ
- เพื่อทำให้หน่วยงานมีการติดต่อสื่อสารได้สะดวกมากขึ้น เนื่องจากระบบเครือข่ายได้ทำการเชื่อมโยงจุดต่างๆให้สามารถสื่อสารกันได้ ดังนั้นทำให้สะดวกในการรับ-ส่งแฟ้มข้อมูล หรือส่งข้อความระหว่างจุดต่างๆ ในเครือข่าย

## 2.3 แบบแผนของการกระจายข้อมูล (Data distribution strategies) [8]

องค์ประกอบที่สำคัญสำหรับการกำหนดแบบแผน การกระจายของข้อมูล คือ จำนวนที่ตั้งแหล่ง

ข้อมูล และปริมาณของข้อมูลที่มีการทำสำเนา โดยแบบแผนการกระจายสามารถกำหนดได้ในรูปแบบต่างๆ กัน ดังนี้

2.3.1 แบบศูนย์กลาง (Centralized) เป็นแบบที่แต่ละจุดจะเชื่อมต่อมาเพื่อขอข้อมูลที่ส่วนกลาง

2.3.2 แบบจัดแบ่ง (Partitioned) เป็นแบบที่แต่ละจุดจะเก็บเฉพาะข้อมูลของตนเอง โดยไม่มีการเก็บข้อมูลที่ซ้ำซ้อนกัน

2.3.3 แบบสำเนา (Replicated) เป็นแบบที่หลายๆ ฐานข้อมูลมีการทำ สำเนาข้อมูลมาเก็บที่ฐานข้อมูลของตนเอง

2.3.4 แบบผสม (Hybrid) เป็นแบบที่ฐานข้อมูลแต่ละแห่งประกอบด้วย แบบสำเนาข้อมูลและแบบจัดแบ่ง บนฐานข้อมูลของตนเอง

จากรูปแบบต่างๆ ของการกระจายข้อมูลดังกล่าว สามารถมองได้เป็น 2 แบบหลักคือ แบบเก็บข้อมูลไม่ซ้ำซ้อน (แบบศูนย์กลาง และแบบจัดแบ่ง) กับ แบบเก็บข้อมูลซ้ำซ้อน (แบบสำเนา และแบบผสม) โดยในที่นี้จะกล่าวถึงแบบเก็บข้อมูลซ้ำซ้อน ซึ่งประกอบด้วยแบบจำลอง ที่ทำให้ข้อมูลมีความถูกต้อง ดังนี้

- แบบจำลองหลัก/รอง (Master/slave model) [5]

แบบจำลองนี้จะมีส่วนที่เป็นแหล่งข้อมูลหลัก ซึ่งเป็นที่เก็บข้อมูลตั้งอยู่ ณ ส่วนกลางหรือกระจายออกไปก็ได้ โดยกลุ่มของตารางมีการจัดแบ่งข้อมูลแต่ละส่วนเพื่อส่งผ่านไปตามจุดสถานที่ต่างๆกัน การส่งผ่านดังกล่าวจะมีเวลาเป็นเงื่อนไขที่สำคัญ ซึ่งจะต้องตระหนักว่า ณ จุดเวลาหนึ่งต้องมีเพียงข้อมูลของแหล่งข้อมูลเดียวเท่านั้น ที่ทำหน้าที่เป็นแหล่งข้อมูลหลัก เพราะหลักการนี้จะทำให้สามารถกำหนดแหล่งข้อมูล ที่จะมีการปรับปรุงข้อมูล

กรณีที่แหล่งข้อมูลหลักเปลี่ยนแปลง การทำสำเนาจะเกิดขึ้นในลักษณะการทำสำเนาแบบไม่ประสานเวลา โดยข้อมูลที่เปลี่ยนแปลง ณ แหล่งข้อมูลหลัก และแหล่งข้อมูลสำเนา จะแตกต่างเหลื่อมกันในช่วงเวลาดังกล่าว กล่าวโดยสรุปสำหรับแบบจำลองนี้ การทำสำเนาข้อมูลจะเกิดหลายๆจุด โดยที่แหล่งข้อมูลหลักทำการเปลี่ยนแปลง แล้วอนุญาตให้ผู้ใช้จากจุดอื่นๆ เข้ามาทำการสอบถามข้อมูล แต่จะไม่อนุญาตให้ทำการเปลี่ยนแปลงข้อมูล โดยข้อมูลที่อ่านได้จะมีความถูกต้องที่ถูกกำหนดด้วยวิธีการทำสำเนาแบบไม่ประสานเวลา

- แบบจำลองปรับปรุงจุดต่างๆ (Update-anywhere model)

แบบจำลองนี้จะไม่มีการระบุแหล่งข้อมูลหลัก แหล่งข้อมูลที่เป็นจุดสำเนาข้อมูลสามารถที่จะทำการเปลี่ยนแปลงข้อมูล ณ แหล่งข้อมูลได้ตลอดเวลา โดยการสำเนานี้สามารถทำได้ทั้งสำเนา

แบบประสานเวลา และสำเนาแบบไม่ประสานเวลา

## 2.4 ฐานข้อมูล ( Database )

ฐานข้อมูลเป็นแหล่งสำหรับเก็บรวบรวมข้อมูลหรือกลุ่มของข้อมูลไว้ด้วยกัน ด้วยการจัดรูปแบบการเก็บภายใต้หลักเกณฑ์ เดียวกัน และมีความสัมพันธ์กัน เพื่อให้สามารถนำข้อมูล มาใช้ให้เกิดประโยชน์และมีความถูกต้อง [15]

ฐานข้อมูล ถูกออกแบบมาเพื่อให้เกิดประโยชน์ด้านต่างๆ ดังนี้

- เพื่อผู้ใช้จะได้รับความสะดวกในการใช้ข้อมูลของระบบ
- เพื่อจัดการกับข้อมูลที่มาอยู่ร่วมกัน
- เพื่อควบคุมการเก็บข้อมูลและวิธีการในการเข้าถึงข้อมูลสำหรับนำไปใช้งาน
- เพื่อเพิ่มประสิทธิภาพในด้านการปฏิบัติงานและสามารถซ่อมบำรุงได้

## 2.5 แบบจำลองเชิงสัมพันธ์ ( Relational Data Model )

แบบจำลองเชิงสัมพันธ์ เป็นแนวคิดที่ใช้อธิบายโครงสร้างของฐานข้อมูล ในด้านประเภทข้อมูลและความสัมพันธ์ในลักษณะการเก็บข้อมูลในรูปตารางซึ่งตารางดังกล่าว คือ ความสัมพันธ์ ( Relation ) ที่มี 2 มิติ คือ แถว( Row ) ซึ่งเปรียบเสมือนระเบียนของข้อมูล และมีคอลัมน์ ( Column ) เปรียบได้กับเขตข้อมูล

## 2.6 ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ ( Relational Database Management System : RDBMS )

ระบบจัดการฐานข้อมูลเชิงสัมพันธ์เป็น ระบบการจัดการฐานข้อมูลที่ออกแบบฐานข้อมูลโดยใช้หลักแบบจำลองเชิงสัมพันธ์ โดยจะต้องมีลักษณะการจัดการกับข้อมูล ตามกฎของคอดด์ (Codd ‘s Rules) ดังต่อไปนี้ [19]

กฎข้อที่ 0 ระบบจัดการฐานข้อมูลแบบเชิงสัมพันธ์ ต้องสามารถจัดการกับข้อมูลที่เก็บอยู่ในฐานข้อมูลด้วยความสามารถเชิงสัมพันธ์

กฎข้อที่ 1 ข้อมูลทุกอย่างที่เก็บในฐานข้อมูลแบบเชิงสัมพันธ์ จะต้องเก็บในลักษณะของตารางเท่านั้น

กฎข้อที่ 2 ต้องสามารถเข้าถึงข้อมูลต่างๆ ในตารางโดยการระบุเพียงชื่อตาราง ชื่อคอลัมน์ และค่าของกุญแจหลักเท่านั้น

กฎข้อที่ 3 ค่าว่าง (NULL) จะต้องมีความหมายแตกต่างไปจากช่องว่าง หรือ ตัวอักษรความยาวศูนย์ และค่าตัวเลขศูนย์ เพื่อให้เห็นความแตกต่างระหว่างข้อมูลที่ยังไม่มีการบันทึก กับข้อมูลที่ไม่มีค่า

กฎข้อที่ 4 สารบัญแฟ้ม (Catalog) ซึ่งเป็นส่วนที่เก็บข้อมูลเกี่ยวกับโครงสร้างของฐานข้อมูล จะต้องเก็บในลักษณะของตาราง และจะต้องสามารถจัดการได้เช่นเดียวกับข้อมูลที่เก็บในฐานข้อมูล

กฎข้อที่ 5 จะต้องมีภาษาที่ใช้ควบคุมในการจัดการฐานข้อมูล กำหนดโครงสร้างของข้อมูล การรักษาความบูรณภาพของข้อมูล ในการจัดการข้อมูล ซึ่งในปัจจุบันระบบจัดการฐานข้อมูลจะใช้ภาษาสอบถามเชิงโครงสร้าง ( Structures Query Language: SQL)

กฎข้อที่ 6 การเปลี่ยนแปลงข้อมูลในตารางที่กำหนดตามมุมมองของผู้ใช้ (User view) ระบบจัดการฐานข้อมูล จะต้องแก้ไขที่ฐานข้อมูลจริงด้วย

กฎข้อที่ 7 ผู้ใช้สามารถที่จะเรียกตรวจสอบข้อมูล เพิ่มเติม ปรับปรุง หรือลบข้อมูลในระบบฐานข้อมูล ด้วยคำสั่งเพียงหนึ่งคำสั่ง

กฎข้อที่ 8 ผู้ใช้ไม่จำเป็นต้องทราบวิธีการจัดเก็บข้อมูลในอุปกรณ์ และถ้ามีการเปลี่ยนแปลงรูปแบบการจัดเก็บจะต้องไม่กระทบกระเทือนต่อโปรแกรมที่ใช้งานซึ่งอยู่ในระดับสูงกว่า

กฎข้อที่ 9 ระบบจัดการฐานข้อมูล ต้องมีความอิสระของข้อมูลในระดับตรรกะ โดยการเปลี่ยนแปลงใดๆ ในส่วนนี้จะไม่ส่งผลกระทบต่อโปรแกรมใช้งาน ซึ่งอยู่ในระดับสูงกว่า

กฎข้อที่ 10 กฎต่างๆ ที่ใช้สำหรับความถูกต้อง ความบูรณภาพของข้อมูลในระบบฐานข้อมูลจะต้องกำหนดได้โดยใช้ภาษาของระบบ และต้องเก็บข้อกำหนดเหล่านี้อยู่ในสารบัญแฟ้มของระบบ โดยมีความเป็นอิสระในการจัดเก็บข้อมูล ไม่ขึ้นกับโปรแกรมหรือ ระดับสูงกว่า

กฎข้อที่ 11 ระบบจัดการฐานข้อมูล จะต้องมีความเป็นอิสระต่อการกระจายข้อมูล โดยข้อมูลที่อยู่กระจัดกระจาย เมื่อมีการเคลื่อนย้ายข้อมูลภายใน เครือข่ายจะต้องไม่มีผลกระทบต่อโปรแกรมใช้งาน

กฎข้อที่ 12 การเข้าถึงข้อมูลโดยตรงอนุญาตให้กระทำได้เฉพาะ คำสั่งที่ใช้เรียกข้อมูลจากฐานข้อมูลเท่านั้น โดยไม่ได้เรียกใช้ผ่านโปรแกรมประยุกต์และข้อมูลที่ส่งมาเก็บที่ฐานข้อมูลโดยผ่านโปรแกรมชุดดังกล่าวจะต้องถูกตรวจสอบความถูกต้องทุกครั้ง

## 2.7 ฐานข้อมูลแบบกระจาย (Distributed database)

ระบบฐานข้อมูลที่มีฐานข้อมูลเก็บอยู่ในระบบคอมพิวเตอร์หลายๆ เครื่อง โดยติดตั้งอยู่ตามที่ต่างๆ กัน แต่คอมพิวเตอร์เหล่านี้สามารถติดต่อสื่อสารกันได้ ซึ่งผู้ใช้สามารถเรียกใช้ข้อมูลจากฐานข้อมูลที่อยู่ที่เครื่องใดๆ ก็ได้ โดยผู้ใช้ไม่จำเป็นต้องทราบว่าข้อมูลที่ตนต้องการนั้นจัดเก็บอยู่บนเครื่องใด รูปแบบดังกล่าวนี้เป็นการจัดการฐานข้อมูล ในลักษณะ ฐานข้อมูลแบบกระจาย

## 2.8 ระบบจัดการฐานข้อมูลแบบกระจาย (Distributed Database Management System)

ระบบจัดการฐานข้อมูลแบบกระจายจะต้องทำหน้าที่ในการจัดการระบบที่มีฐานข้อมูลติดตั้งในลักษณะของ ฐานข้อมูลแบบกระจาย โดยระบบจัดการฐานข้อมูล ดังกล่าว ต้องมีลักษณะดังนี้ [1]

- แหล่งข้อมูลในแต่ละแห่งในระบบฐานข้อมูลแบบกระจาย จะต้องเป็นอิสระกัน (Local autonomy)
- ระบบฐานข้อมูลที่กระจายออกมา จะต้องไม่ไปอาศัยฐานข้อมูลที่ส่วนกลางเป็นหลัก ในการทำงาน (No reliance on a central site)
- ระบบฐานข้อมูลที่กระจายออกมา จะต้องทำงานได้ตลอดเวลาอย่างต่อเนื่อง (Continuous operation)
- ผู้ใช้ไม่ต้องทราบตำแหน่งที่ตั้งของระบบฐานข้อมูล ในขณะที่ใช้งานฐานข้อมูล (Location transparency and location independence)
- ผู้ใช้ไม่ต้องทราบว่าตารางต่างๆ ที่ใช้งานอยู่ขณะนั้นถูกแยกไปเก็บในระบบฐานข้อมูลใดบ้าง (Fragmentation independence)
- การทำสำเนาจะทำให้ข้อมูลชุดเดียวกัน สามารถที่จะกระจายไปเก็บที่ฐานข้อมูลต่างๆ ได้ (Replication independence)
- การประมวลผลคำสั่ง ที่ใช้ในการสอบถามข้อมูลต้องเกิดขึ้น ณ เครื่องที่เป็นแหล่งข้อมูล (Distributed query processing)
- ลักษณะของระบบแบบกระจายดังกล่าว ต้องรองรับการทำงานในลักษณะหน่วยรายการแบบไกล และแบบหน่วยรายการแบบกระจาย
- การใช้ข้อมูลของระบบฐานข้อมูล ที่กระจายอยู่บนระบบฐานข้อมูลแบบกระจายต้องไม่ขึ้นอยู่กับลักษณะของฮาร์ดแวร์ที่ใช้งาน(Hardware independence)

- การติดต่อระหว่างฐานข้อมูล บนระบบฐานข้อมูลแบบกระจาย ต้องสามารถทำได้โดยไม่ขึ้นกับระบบปฏิบัติการที่ใช้ (Operating system independence)
- การติดต่อระหว่างฐานข้อมูล บนระบบฐานข้อมูลแบบกระจายต้องสามารถทำได้โดยไม่ขึ้นกับโพรโทคอล ของเครือข่าย (Network independence)
- ระบบจัดการฐานข้อมูลที่แตกต่างกัน (Heterogeneous DBMS) ตามแหล่งข้อมูลต่างๆ จะต้องสามารถทำงานร่วมกันได้ (DBMS independence)

ระบบจัดการฐานข้อมูลแบบกระจาย ทำให้ผู้ใช้ที่เป็นเจ้าของข้อมูลสามารถจัดการและควบคุมการใช้ข้อมูลได้อย่างเต็มที่ การกำหนดสิทธิและดูแลความปลอดภัยของข้อมูลเป็นไปได้อย่างสะดวก ผู้ใช้ที่ทำงานกับเฉพาะข้อมูลของตนเองนี้จะได้รับผลลัพธ์การทำงานที่รวดเร็ว เนื่องจากไม่ต้องรอการใช้ข้อมูลจากผู้ใช้จากจุดอื่นหรือปัญหาของเครือข่ายที่เชื่อมต่อ นอกจากนี้กรณีที่ฐานข้อมูลมีขนาดใหญ่ขึ้นการขยายระบบคอมพิวเตอร์เพื่อให้รองรับปริมาณงานที่มากขึ้นจะทำได้ง่ายเพราะ ไม่ได้ส่งผลกระทบกระเทือนกับงานของส่วนรวม นอกจากนี้ถ้าคอมพิวเตอร์เกิดความเสียหายจะเกิดผลกับผู้ใช้ที่ใช้ข้อมูลที่จัดเก็บอยู่ในเครื่องที่เสียหาย เท่านั้น โดยส่วนรวมแล้วผู้ใช้ ณ จุดอื่นจะสามารถใช้งานได้ปกติ

อย่างไรก็ตาม ระบบจัดการฐานข้อมูลแบบกระจายจะทำให้ผู้ออกแบบฐานข้อมูลจำเป็นที่จะต้องมีการออกแบบการเก็บข้อมูลบางส่วนที่ซ้ำกันสำหรับบางจุด เพราะกรณีที่ข้อมูลเกิดความเสียหายขึ้นที่จุดหนึ่ง ผู้ใช้ก็จะสามารถใช้ข้อมูลดังกล่าว จากจุดอื่นๆ ที่ไม่ได้รับความเสียหายได้ ถ้าผู้ใช้ต้องการแก้ไขปรับปรุงข้อมูลที่มีการเก็บซ้ำซ้อนจะทำได้ยากเพราะจะต้องมีการปรับปรุงข้อมูลให้ครบทุกๆจุด สำหรับข้อมูลชุดนี้ถ้าต้องการปรับปรุงพร้อมๆ กันทุกจุด จะต้องมีการควบคุมภาวะความพร้อมกันและต้องคืนข้อมูลเดิมกลับมาทันทีที่การปรับปรุงทำไม่สำเร็จ ซึ่งถ้าผู้ใช้ให้การยอมรับความถูกต้องของข้อมูลได้ที่ระดับหนึ่ง โดยไม่ต้องการให้การปรับปรุงข้อมูลซับซ้อนมากนัก การทำสำเนาข้อมูลจะสามารถนำมาใช้เพื่อแก้ไขปัญหา ดังกล่าวได้

ลักษณะของระบบจัดการฐานข้อมูลแบบกระจายที่นิยามข้างต้น รวมถึงลักษณะบางประการที่ผู้ใช้จะต้องคำนึงถึง ผู้วิจัยต้องการชี้ให้เห็นถึงคุณสมบัติเหล่านี้ เพื่อที่จะเป็นแนวทางที่ต้องใช้ในการกล่าวถึงความเป็นอิสระในการทำการสำเนาข้อมูล (Replication independence) ต่อไป

## 2.9 งานวิจัยที่เกี่ยวข้อง

งานวิจัย "A STUDY OF A DISTRIBUTED DATA PROCESSING SYSTEM"[4]

เป็นการศึกษาลักษณะการประมวลผลแบบกระจาย (Distributed Data Processing ; DDP) ผู้วิจัยได้อธิบายถึงลักษณะของการประมวลผล แบบออนไลน์ที่แบ่งย่อยได้เป็น 2 แบบคือ การประมวลผลแบบรวมศูนย์ (Centralized) และการประมวลผล แบบกระจาย (Decentralized) ผู้วิจัยกล่าวถึงปัญหาต่างๆ ที่เกิดขึ้นทำให้ผู้วิจัยได้นำลักษณะการประมวลผลแบบกระจาย มาศึกษาถึงปัญหาและอุปสรรคต่างๆที่อาจจะเกิดขึ้น จากการนำระบบดังกล่าวมาใช้งาน ผู้วิจัยได้แบ่งการวิเคราะห์ตามโครงสร้างการสื่อสารและฐานข้อมูล โดยเฉพาะในส่วนของฐานข้อมูลรูปแบบหนึ่งของการประมวลผลแบบกระจายจะเป็นแบบ Replicated Database

ผู้วิจัยได้กล่าวถึงองค์กรราชการและเอกชนที่สามารถนำเอารูปแบบการประมวลผลแบบกระจายไปใช้งานในรูปแบบต่างๆ

## 2.10 การสำเนาข้อมูล

หมายถึง การที่แหล่งข้อมูลแห่งหนึ่ง มีการทำสำเนาข้อมูลจากแหล่งข้อมูลหลัก ทำให้มีข้อมูลส่วนดังกล่าว เหมือนแหล่งข้อมูลหลัก นอกจากการทำสำเนาข้อมูลแล้วจะรวมถึง การวิเคราะห์ ออกแบบ ติดตั้ง ดูแลระบบและการเฝ้าติดตามตรวจสอบ เพื่อให้การส่งผ่านข้อมูลระหว่างแหล่งข้อมูลถูกต้อง การติดตั้งการสำเนาข้อมูล มี 2 ประเภท ดังนี้ [1]

### 2.10.1 การทำสำเนาข้อมูลแบบ ประสานเวลา

วัตถุประสงค์ของการทำสำเนาข้อมูลแบบนี้ เพื่อต้องการให้ข้อมูลระหว่างแหล่งข้อมูลใดๆที่ทำสำเนาข้อมูลเป็นข้อมูลเดียวกันชุดเดียวกันและถูกต้องมากที่สุด (tight consistency) ซึ่งต้องอาศัยการยืนยันรายการแบบ 2 ขั้นตอน สำหรับลักษณะงานดังกล่าว

สำหรับการทำสำเนาแบบนี้ เหมาะกับลักษณะระบบที่ต้องการความทันสมัยถูกต้องของข้อมูลมากที่สุด เพราะขณะที่ข้อมูลบนแหล่งข้อมูลแห่งใดมีการปรับปรุง ณ เวลานั้นแหล่งข้อมูลที่จุดอื่นๆจะต้องถูกปรับปรุงไปพร้อมกันให้มีลักษณะของข้อมูลเหมือนกัน ด้วยวิธีการบริหารระบบแบบการกระจายรายการ ดังนั้นจะต้องมีการยืนยันรายการทั้งหมด (global commit) แต่สำหรับลักษณะระบบงานที่ยอมรับได้กับความถูกต้องที่ระดับหนึ่งซึ่งไม่ได้คำนึงว่าข้อมูลจะต้องทันสมัยตลอดเวลา ผู้ทำงานวิจัยเสนอให้ใช้รูปแบบการทำสำเนา ที่จะกล่าวต่อไป

2.10.2 การทำสำเนาข้อมูลแบบ ไม่ประสานเวลา

วัตถุประสงค์ของการทำสำเนาข้อมูลแบบนี้ เพื่อต้องการให้ข้อมูลระหว่างแหล่งข้อมูลใดๆที่ทำสำเนาข้อมูล เป็นข้อมูลเดียวกันชุดเดียวกัน แต่ไม่คำนึงถึงความถูกต้องของข้อมูล ณ เวลาใดๆ มากนัก กล่าวคือที่เวลาหนึ่งความไม่ตรงกัน ของข้อมูลชุดเดียวกันบนแหล่งข้อมูลต่างๆ จะเกิดขึ้นโดยการทำสำเนาวิธีนี้ เมื่อมีการยืนยันรายการที่แหล่งข้อมูลหลักเรียบร้อยแล้ว ต้องอาศัยเวลาอีกช่วงหนึ่ง รายการดังกล่าวจึงจะถูกส่งผ่านไปยังจุดอื่นๆ เพื่อทำการปรับปรุง การทำสำเนานี้สามารถติดตั้งได้เป็นรูปแบบต่างๆ ดังนี้

- การนำมาเพิ่มแบบสมบูรณ์หรือแบบเพิ่มเติม (Complete or incremental refresh)

สำหรับการนำมาเพิ่มแบบสมบูรณ์ การอ่านข้อมูลทั้งหมดจากแหล่งข้อมูลจะทำตามตารางเวลาที่ได้ตั้งไว้ ปลายทางที่ข้อมูลถูกส่งไปจะทำการสำเนาข้อมูลไว้ สำหรับการนำมาเพิ่มแบบเพิ่มเติม ขั้นตอนจะเหมือนกับแบบแรกแต่ต่างกันที่ การส่งผ่านข้อมูลจะส่งเฉพาะข้อมูล ที่มีการเปลี่ยนแปลงเท่านั้น และพบว่าการประมวลผลในส่วนนี้ จะเป็นลักษณะการประมวลผลแบบเป็นกลุ่ม (batch processing) เพราะฉะนั้นข้อมูลที่แหล่งข้อมูล ที่เปลี่ยนแปลงจะถูกเก็บสะสมไว้ ในลักษณะเข้าแถวคอย (queuing) ตามลำดับ (staging) เพื่อรอเวลาในการส่งผ่านไปเปลี่ยนแปลง ณ จุดที่เป็นเป้าหมาย โดยข้อมูลจะถูกนำเข้าด้วยโปรแกรมอรรถประโยชน์ ซึ่งจะทำเสมือนเป็นผู้ใช้คนหนึ่งในระบบด้วย ดังนั้นในขณะที่ข้อมูลนั้นยังทำการเปลี่ยนแปลงไม่เรียบร้อย ผู้ใช้คนถัดมาที่เข้ามาขอใช้ข้อมูลชุดดังกล่าว จะยังไม่สามารถใช้ได้จนกว่าการเปลี่ยนแปลงจะสมบูรณ์เรียบร้อยแล้ว

- การสำเนาข้อมูลระหว่างฐานข้อมูลกับฐานข้อมูล (Database-to-database asynchronous replication) เป็นการสำเนาข้อมูลที่จัดทำขึ้น จากผู้ผลิตซอฟต์แวร์ด้านฐานข้อมูล โดยผู้ผลิตแต่ละรายจะมีเทคนิควิธีการของตนเอง ในการทำสำเนาข้อมูลซึ่งจะมีลักษณะเฉพาะ ดังนี้ [1]

  1) เป็นการทำแบบตารางกับตาราง (Table-to-table event-based replication)
  2) เป็นการทำระดับแถว (Rows level) เช่น ระบบฐานข้อมูล ออราเคิล (Oracle) [9,12] ระบบฐานข้อมูล เอสคิวแอลเซิฟเวอร์ ( SQL Server) [11] และ ระบบฐานข้อมูล ไซเบส (Sybase) [10]
  3) การจัดเรียงลำดับรายการที่เกิดที่แหล่งข้อมูล แล้วให้กระทำที่ปลายทางเพื่อจะทำการควบคุมระดับของความถูกต้อง

- การสำเนาระหว่างโปรเซสกับโปรเซส (Process-to-process asynchronous

replication) สำหรับการทำสำเนาในลักษณะนี้จะอยู่บนฐานของรูปแบบการส่งข้อมูลซับสไคร์บเชื่อมต่อคุยกัน ถ้ามีหลายชั้น (multitier) ทำให้การทำสำเนาแบบนี้ติดตั้งเป็นชั้นกลาง (middle tier) โดยมีองค์ประกอบหลัก ดังนี้คือ

1) โปรเซสจะคอยตรวจว่า มีเหตุการณ์ใดเกิดขึ้นที่ตรงกับรูปแบบที่ได้กำหนดไว้ว่า ให้มีการทำสำเนาเกิดขึ้น
2) เมื่อตรวจพบเหตุการณ์ที่ต้องทำสำเนาจะทำการเก็บข้อมูลที่จะไปเปลี่ยนแปลงที่ปลายทางไว้ และเตรียมส่งออกไปด้วยการใช้แบบจำลอง เก็บไว้แล้วทำการส่งต่อ (store-and-forward paradigm)
3) โปรเซสบริการที่อยู่ในรูปของโปรแกรมประยุกต์ จะนำรายการที่พร้อมส่งต่อจากแหล่งข้อมูลหลัก มายังเครื่องเป้าหมาย
4) โปรเซสที่มีหน้าที่นำข้อมูลที่เปลี่ยนแปลงที่แหล่งข้อมูลหลัก นำมาเปลี่ยนแปลงที่จุดของแหล่งข้อมูลที่ทำสำเนา

ผู้วิจัยพบว่า การทำสำเนาแบบนี้มีลักษณะที่น่าศึกษา คือความสามารถในการทำงานร่วมกับรูปแบบอื่นๆ ที่ไม่ได้เป็นระบบฐานข้อมูล เช่น รูปแบบรายการโคบอล(COBOL record) แฟ้มข้อมูลบนระบบยูนิกซ์ แฟ้มข้อมูลที่มีรูปแบบแน่นอน รวมถึงแฟ้มข้อมูลที่มีรูปแบบผันแปร จะสามารถใช้วิธีนี้ได้

จากที่กล่าวมาข้างต้น กรณีที่ความต้องการของระบบต้องการข้อมูลที่ถูกต้องโดยไม่คำนึงว่าข้อมูลที่ใช้จะต้องมีค่าเดียวกันกับแหล่งข้อมูลหลักตลอดเวลาและยอมรับความถูกต้องข้อมูลได้ในระยะเวลาหนึ่ง การทำสำเนาข้อมูลแบบไม่ประสานเวลา จะเป็นวิธีที่ดีที่สุด [1] ผู้วิจัยได้วิเคราะห์รูปแบบการทำสำเนาข้อมูลต่างๆ ที่กล่าวมาข้างต้น เพื่อเป็นแนวทางในการทำวิจัยที่จะกล่าวต่อไป