

แผนการสอนประจำบทเรียน

รายชื่ออาจารย์ผู้จัดทำ ลัษณา อินชัยวงศ์

รายละเอียดของเนื้อหา

ตอนที่ 10.1 แนวคิดเรื่องบูรณภาพของข้อมูล

เรื่องที่ 10.1.1 ความหมายและชนิดของบูรณภาพของข้อมูล

เรื่องที่ 10.1.2 กฎเกณฑ์ของข้อมูล และวิธีการจัดการกับข้อมูลเพื่อให้เป็นไปตามกฎเกณฑ์

ตอนที่ 10.2 การประมวลผลรายการเปลี่ยนแปลงเบื้องต้น

เรื่องที่ 10.2.1 แนวคิดเรื่องระบบและการประมวลผลรายการเปลี่ยนแปลง

เรื่องที่ 10.2.2 คุณสมบัติของการประมวลผลรายการเปลี่ยนแปลงที่จำเป็น

เรื่องที่ 10.2.3 การจัดลำดับการทำงานของรายการเปลี่ยนแปลง

ตอนที่ 10.3 การจัดการรายการเปลี่ยนแปลง

เรื่องที่ 10.3.1 การกู้คืน

เรื่องที่ 10.3.2 การประมวลผลรายการเปลี่ยนแปลงอย่างเป็นลำดับ

เรื่องที่ 10.3.3 การจัดลำดับการทำงานของรายการเปลี่ยนแปลง

ตอนที่ 10.4 การจัดการรายการเปลี่ยนแปลงขั้นสูง

เรื่องที่ 10.4.1 การตรวจสอบการประมวลผลรายการเปลี่ยนแปลง

เรื่องที่ 10.4.2 ระบบการประมวลผลรายการเปลี่ยนแปลงแบบทันเวลา

แนวคิด

1. บูรณภาพของข้อมูล ทำให้ข้อมูลในระบบฐานข้อมูลมีความถูกต้องอยู่เสมอ การเปลี่ยนแปลงข้อมูลจากผู้ใช้งานข้อมูลหรือโดยระบบจัดการฐานข้อมูลจะต้องยังผลให้ข้อมูลและความสัมพันธ์ระหว่างข้อมูลถูกต้องตามโครงสร้างของฐานข้อมูล
2. ในการรักษาความถูกต้องของข้อมูลในฐานข้อมูลจะต้องเป็นไปตามกฎเกณฑ์ที่ถูกต้อง อันได้แก่ กฎเกณฑ์ในเรื่องความถูกต้องของตัวข้อมูลเอง กฎเกณฑ์ในเรื่องความสัมพันธ์ระหว่างข้อมูล และกฎเกณฑ์ในการประมวลผลรายการเปลี่ยนแปลงในฐานข้อมูล
3. ขั้นตอนการประมวลผลรายการเปลี่ยนแปลงมีผลโดยตรงกับข้อมูลในฐานข้อมูล ดังนั้นจึงจำเป็นต้องแน่ใจว่าผลลัพธ์จากการประมวลผลจะทำให้ข้อมูลมีความถูกต้องอยู่เสมอ

4. รายการเปลี่ยนแปลงจะต้องมีคุณสมบัติครบถ้วนตามที่กำหนดเพื่อรักษาความถูกต้องของข้อมูลเอาไว้
5. การจัดลำดับการทำงานของรายการเปลี่ยนแปลงสามารถทำได้ทั้งแบบเรียงลำดับและเหมือนเรียงลำดับ โดยยึดถือความถูกต้องของผลลัพธ์เป็นหลัก
6. ข้อมูลเป็นสิ่งที่มีความสำคัญมาก ความเสียหายที่ขึ้นกับระบบคอมพิวเตอร์อาจจะมีผลกระทบกับข้อมูลในฐานะข้อมูล ดังนั้นการทำให้ข้อมูลมีความถูกต้องสมบูรณ์ดังเดิม จึงเป็นเรื่องที่จำเป็นมากเช่นเดียวกัน
7. เมื่อทราบสาเหตุที่ทำให้เกิดความเสียหายกับข้อมูล ก็สามารถที่จะหาวิธีแก้ไขและกู้ข้อมูลคืนได้
8. การตรวจสอบความถูกต้องในการประมวลผลรายการเปลี่ยนแปลงจะช่วยรักษาความถูกต้องในการทำงานของระบบจัดการฐานข้อมูล และช่วยรักษาความถูกต้องของข้อมูลภายในฐานข้อมูลเอาไว้ได้
9. การประมวลผลข้อมูลมีหลายประเภท ข้อมูลที่ระยะเวลาในการปรากฏของข้อมูลมีความสำคัญมาก วิธีการประมวลผลที่เลือกใช้ก็ต้องเป็นแบบที่สามารถให้ผลลัพธ์ของการประมวลผลได้ในเวลาที่ต้องการ

วัตถุประสงค์

หลังจากศึกษาบทเรียนที่ 10 แล้ว นักศึกษาสามารถ

1. ให้คำจำกัดความของบูรณภาพของข้อมูลได้
2. เขียนกฎเกณฑ์ความถูกต้องของข้อมูลได้
3. เขียนคุณสมบัติที่จำเป็นของรายการเปลี่ยนแปลงได้
4. เขียนลำดับการประมวลผลรายการเปลี่ยนแปลงที่ถูกต้องได้
5. เรียนรู้และเข้าใจถึงภาวะการเกิดความเสียหายของข้อมูลและวิธีการกู้คืน
6. เขียนถึงวิธีการจัดการกับความเสียหายที่เกิดขึ้นกับข้อมูลได้
7. เรียนรู้วิธีการตรวจสอบความถูกต้องของการประมวลผลรายการเปลี่ยนแปลง
8. บอกชื่อและลักษณะของระบบประมวลผลรายการเปลี่ยนแปลงขั้นสูงได้

กิจกรรมการเรียนรู้การสอน

กิจกรรมที่นักศึกษาต้องทำสำหรับการเรียนการสอน ได้แก่

1. ศึกษาเอกสารการสอน ตอนที่ 10.1, 10.2, 10.3 และตอนที่ 10.4 ได้
2. ทำกิจกรรมในรูปแบบฝึกปฏิบัติบทเรียนที่ 10
3. ทำแบบประเมินผลของบทเรียนที่ 10

สื่อการสอน

เอกสารการสอน

1. เอกสารประกอบการสอน

ประเมินผล

1. ประเมินผลจากกิจกรรมที่ทำ
2. ประเมินผลจากคำถามท้ายบทเรียน

ตอนที่ 10.1 แนวคิดเรื่องบูรณภาพของข้อมูล

หัวเรื่อง

- เรื่องที่ 10.1.1 ความหมายและชนิดของบูรณภาพของข้อมูล
- เรื่องที่ 10.1.2 กฎเกณฑ์ของข้อมูล และวิธีการจัดการกับข้อมูลเพื่อให้เป็นไปตามกฎเกณฑ์

แนวคิด

1. บูรณภาพของข้อมูล ทำให้ข้อมูลในระบบฐานข้อมูลมีความถูกต้องอยู่เสมอ การเปลี่ยนแปลงข้อมูลจากผู้ใช้งานข้อมูลหรือโดยระบบจัดการฐานข้อมูลจะต้องยังผลให้ข้อมูลและความสัมพันธ์ระหว่างข้อมูลถูกต้องตามโครงสร้างของฐานข้อมูล
2. ในการรักษาความถูกต้องของข้อมูลในฐานข้อมูลจะต้องเป็นไปตามกฎเกณฑ์ที่ถูกต้อง อันได้แก่ กฎเกณฑ์ในเรื่องความถูกต้องของตัวข้อมูลเอง กฎเกณฑ์ในเรื่องความสัมพันธ์ระหว่างข้อมูล และกฎเกณฑ์ในการประมวลผลรายการเปลี่ยนแปลงในฐานข้อมูล

วัตถุประสงค์

หลังจากที่ศึกษาตอนที่ 10.1 แล้ว นักศึกษาสามารถ

1. ให้คำจำกัดความของบูรณภาพของข้อมูลได้
2. เขียนกฎเกณฑ์ความถูกต้องของข้อมูลได้

เรื่องที่ 10.1.1 ความหมายและชนิดของบูรณภาพของข้อมูล

บูรณภาพ หมายถึง ความถูกต้อง ความมั่นคง และความเป็นอันหนึ่งอันเดียวกัน ดังนั้น บูรณภาพของข้อมูล จึงหมายความว่า ความถูกต้องของข้อมูลในฐานข้อมูล ซึ่งเป็นใจความสำคัญหลักของเนื้อหาในบทนี้ ความสัมพันธ์หลายชนิดที่ประกอบกันเป็นฐานข้อมูลเชิงสัมพันธ์นั้นมีข้อจำกัดและกฎเกณฑ์จำนวนหนึ่งที่เป็นตัวกำหนดว่าข้อมูลจะสามารถอยู่ในฐานข้อมูลได้เพื่อรักษาความสัมพันธ์ของข้อมูลเชิงสัมพันธ์ไว้ และเป็นการป้องกันความเสียหายที่อาจเกิดกับฐานข้อมูล กฎเกณฑ์และข้อจำกัดจะถูกกล่าวถึงรายละเอียดต่อไปในบทนี้

เรื่องที่ 10.1.2 กฎเกณฑ์ของข้อมูล และวิธีการจัดการกับข้อมูลเพื่อให้เป็นไปตามกฎเกณฑ์

กฎเกณฑ์ของข้อมูล คือ กฎเกณฑ์ที่ใช้ในการรักษาความถูกต้องของข้อมูล มีจุดประสงค์หลัก คือ

- ป้องกันความผิดพลาดที่เกิดจากการเพิ่มข้อมูลลงในฐานข้อมูล
- รักษาความถูกต้องของข้อมูลเมื่อมีการเปลี่ยนแปลงข้อมูลในฐานข้อมูล
- ทำให้ข้อมูลต่างๆ ในฐานข้อมูลมีความถูกต้องตรงกัน
- ทำให้ระบบจัดการฐานข้อมูลสามารถตัดสินใจได้ว่า จะจัดการกับข้อมูล ณ ตำแหน่งต่างๆ ในฐานข้อมูลอย่างไร เช่น เลือกตำแหน่งในการเก็บข้อมูล

กฎเกณฑ์ในการคงคุณภาพของข้อมูลสามารถแบ่งได้ 3 ประเภท ได้แก่

1. กฎเกณฑ์เกี่ยวกับชนิดของข้อมูล (Type constraint)
2. กฎเกณฑ์เกี่ยวกับแอททริบิวของข้อมูล (Attribute constraint)
3. กฎเกณฑ์เกี่ยวกับฐานข้อมูล (Database constraint)

10.2.1 กฎเกณฑ์เกี่ยวกับชนิดของข้อมูล (Type constraint)

ข้อมูลแต่ละชนิดในฐานข้อมูลมีลักษณะการเก็บในฐานข้อมูลและการนำไปใช้งานต่างกัน เพื่อให้ข้อมูลเหล่านี้มีความถูกต้องก่อนการนำไปใช้งานจึงต้องมีการตรวจสอบชนิดและค่าของข้อมูลชนิดนั้นว่าถูกต้องหรือไม่ ตัวอย่าง เช่น ในฐานข้อมูลนักศึกษา ชื่อนักศึกษา (student-name) และ ชื่อธนาคาร (bank-name) อาจจะเป็นข้อมูลประเภทเดียวกัน คือ ข้อความ แต่ขอบเขตของ (domain) ข้อมูลต่างกัน ตัวอย่างในการกำหนดขอบเขตสำหรับข้อมูล ชื่อนักศึกษา และ ชื่อธนาคาร โดยใช้ SQL-92 แสดงได้ในบรรทัดต่อไปนี้

```
create domain student-name char(40)
```

```
constraint student-name-constraint check ( value not null )
```

ประโยค create domain ใช้ในการระบุของข้อมูล student-name ให้เป็นชนิด char ซึ่งมีขนาด 40 ตัวอักษร และกำหนดให้ข้อมูลต้องมีค่าอยู่เสมอ ไม่สามารถเป็นค่าว่าง (Null) ได้ โดยใช้ประโยค constraint ตามด้วยการกำหนดขอบเขตของข้อมูลโดยใช้ check

```
create domain bank-name char(20)
```

```
constraint bank-name-constraint check ( value in ( "Siam Commercial Bank", "Krung Thai Bank", "Thai Farmer Bank" ) )
```

จากตัวอย่างทั้งสอง จะเห็นได้ว่าข้อมูลประเภทเดียวกัน ไม่จำเป็นต้องมีขอบเขตเหมือนกัน ทั้งนี้เพื่อประโยชน์ในการตรวจสอบข้อมูลก่อนการเก็บลงในฐานข้อมูล ดังนั้นการปล่อยให้ข้อมูลชื่อนักศึกษาว่างไว้ แต่มีข้อมูลอื่นสำหรับชื่อที่ว่างไว้ หรือ การใส่ชื่อธนาคารเป็นชื่ออื่น นอกจาก "Siam Commercial Bank", "Krung Thai Bank" หรือ "Thai Farmer Bank" เป็นการละเมิดกฎเกณฑ์และขอบเขตที่ตั้งขึ้นสำหรับข้อมูลชื่อนักศึกษา และชื่อธนาคาร เมื่อไม่เป็นไปตามกฎเกณฑ์ข้อมูลที่เกิดจากความผิดพลาดนี้จะไม่

ถูกนำไปเก็บยังฐานข้อมูล และระบบจัดการฐานข้อมูลจะแสดงข้อผิดพลาดนี้ออกมาในชื่อของกฎเกณฑ์ที่ตั้งขึ้นในตอนแรก

10.2.2 กฎเกณฑ์เกี่ยวกับแอททริบิวของข้อมูล (Attribute constraint)

แอททริบิวของข้อมูล (attribute) คือ ข้อมูลย่อยที่ประกอบกันเป็นข้อมูลของสิ่งๆ หนึ่งในฐานข้อมูล เช่น ในฐานข้อมูลนักศึกษาประกอบไปด้วย ตารางประวัตินักศึกษา ตารางการลงทะเบียน และภายในตาราง ก็มีแอททริบิวของตารางนั้นอยู่ เช่น ตารางประวัตินักศึกษา ประกอบไปด้วยแอททริบิว 6 อย่าง คือ รหัสนักศึกษา, ชื่อ, ที่อยู่, วันเกิด, เบอร์โทรศัพท์ และ เลขประจำตัวบัตรประชาชน ส่วนตารางการลงทะเบียน ประกอบไปด้วยแอททริบิว 4 อย่าง คือ รหัสนักศึกษา, ภาควิชา, รหัสวิชา และ เกรด

กฎเกณฑ์ที่ใช้ในการกำหนดความถูกต้องของแอททริบิวข้อมูลสามารถแยกได้ 3 ประเภท ตามชนิดของแอททริบิว คือ

1. บุรณภาพของคีย์ (Key integrity) กล่าวถึง การที่ค่าของคีย์จะต้องเป็นค่าที่มีเอกลักษณ์ ไม่ซ้ำกับข้อมูลใดในแถวอื่น
2. บุรณภาพของเอนทิตี (Entity integrity) กล่าวถึง การที่ค่าของแอททริบิวที่เป็นคีย์หลักไม่สามารถเป็นค่าว่างได้
3. บุรณภาพของการอ้างอิงแอททริบิว (Referential integrity) กล่าวถึง การอ้างอิงถึงแอททริบิวจากความสัมพันธ์หนึ่งในความสัมพันธ์ใด จะต้องเป็นการอ้างอิงถึงแอททริบิวที่มีอยู่จริงในความสัมพันธ์นั้น การอ้างอิงถึงแอททริบิวที่ไม่มีอยู่จริงจะทำให้ไม่สามารถรักษาบุรณภาพของข้อมูลไว้ได้

บุรณภาพของคีย์

คีย์

สมมติให้ D ตารางในฐานข้อมูลเชิงสัมพันธ์ที่ประกอบไปด้วยกลุ่มของแอททริบิวทั้งหมด $A = (A_1, A_2, \dots, A_n)$ กล่าวได้จะกลุ่มของแอททริบิวทั้งหมด A ที่ประกอบขึ้นเป็นความสัมพันธ์หนึ่งจะมีความเป็นเอกลักษณ์ (uniqueness property) คือ ไม่ซ้ำกับความสัมพันธ์ (tuple) ใดใน D เมื่อกลุ่มของแอททริบิวทั้งหมดมีความเป็นเอกลักษณ์แล้วกล่าวได้ว่า กลุ่มย่อย (subset) K ของแอททริบิว A จะมีความเป็นเอกลักษณ์ด้วย กลุ่มย่อยของแอททริบิวนี้เรียกว่า ซุปเปอร์คีย์ ซึ่งประกอบด้วยคีย์รองหลายๆตัวมารวมกัน และจะไม่มีการซ้ำของคีย์รองใดใน K ที่มีเอกลักษณ์อีก

คีย์หลัก

ในหนึ่งตารางของความสัมพันธ์ประกอบไปด้วยกลุ่มของแอททริบิว (attribute) และแอททริบิวบางตัวทำหน้าที่เป็นคีย์ ในกลุ่มคีย์นี้จะมีคีย์อยู่ตัวหนึ่งที่เป็นคีย์หลัก และคีย์ที่เหลือจะทำหน้าที่เป็นคีย์รอง

คีย์นอก

คีย์นอก คือ คีย์ที่ค่าของมันอ้างอิงถึงค่าของแอททริบิวต์ตรงกันในความสัมพันธ์อื่น(ตาราง) ปัญหาที่เกิดขึ้นในฐานข้อมูลเชิงสัมพันธ์จากการใช้คีย์นอกก็คือ บำรุงรักษาของการอ้างอิงแอททริบิวต์ (Referential integrity) ซึ่งจะถูกล่วงถึงต่อไปในหัวข้อ บำรุงรักษาของการอ้างอิงแอททริบิวต์

บำรุงรักษาของเอ็นทิตี

บำรุงรักษาของเอ็นทิตี คือ การกำหนดค่าที่ถูกต้องและอยู่ในขอบเขตให้กับคีย์หลัก นั่นคือ คีย์หลักไม่สามารถเป็นค่าว่าง (Null) ได้ เช่น รหัสนักศึกษา จะต้องถูกกำหนดค่าที่ถูกต้องตามหลักการกำหนดค่าทุกครั้งที่มีการเพิ่มนักศึกษาคนใหม่เข้ามาในฐานข้อมูล มิฉะนั้นจะทำให้ผิดกฎเกณฑ์ของบำรุงรักษาไป

บำรุงรักษาของการอ้างอิงแอททริบิวต์

การอ้างอิงถึงแอททริบิวต์จากความสัมพันธ์อื่น ระบบจัดการฐานข้อมูลจะต้องตรวจสอบให้แน่ใจว่าแอททริบิวต์มีค่าอยู่จริงในความสัมพันธ์นั้น มิฉะนั้นจะทำให้เกิดการอ้างอิงที่ไม่มีอยู่จริง ตัวอย่างเช่น ในความสัมพันธ์ “บัญชีธนาคาร” มีการอ้างอิงถึงธนาคารสาขา “สุเทพ” แต่ในความสัมพันธ์ “สาขานาการ” ไม่ปรากฏว่ามีธนาคารสาขา “สุเทพ” นี้อยู่จริง ดังนั้นระบบจัดการฐานข้อมูลที่ดีจะต้องสามารถตรวจสอบได้ว่าการอ้างอิงแอททริบิวต์จากความสัมพันธ์อื่นดังเช่นในตัวอย่างนี้ถูกต้องหรือไม่ และหาวิธีในการจัดการที่เหมาะสมต่อไป เช่น รายงานให้ผู้ใช้งานฐานข้อมูลทราบถึงความผิดพลาดที่เกิดขึ้น

10.2.3 กฎเกณฑ์เกี่ยวกับฐานข้อมูล (Database constraint)

การประมวลผลที่เกิดขึ้นภายในฐานข้อมูล อันได้แก่ การอ่านข้อมูลและการเขียนข้อมูล มีผลโดยตรงต่อบำรุงรักษาของข้อมูล ผลลัพธ์จากการทำงานบางอย่างของฐานข้อมูลจะทำให้เกิดความเสียหายกับข้อมูลได้ หรืออาจจะทำให้ระบบล้ม ซึ่งจะมีผลให้ฐานข้อมูลอยู่ในสถานะที่ไม่มั่นคง (Unconsistency) ดังนั้นระบบจัดการฐานข้อมูลจึงจำเป็นต้องรักษากฎเกณฑ์ที่เกี่ยวกับการทำงานในฐานข้อมูลเอาไว้ เพื่อป้องกันความเสียหายที่อาจจะเกิดขึ้นได้เหล่านี้ กฎเกณฑ์ที่จำเป็นสำหรับฐานข้อมูลได้แก่

1. คุณสมบัติที่จำเป็นสำหรับรายการเปลี่ยนแปลง

รายการเปลี่ยนแปลงที่เกิดขึ้นกับฐานข้อมูลจะต้องมีคุณสมบัติครบถ้วน เพื่อรักษาบำรุงรักษาของฐานข้อมูลเอาไว้ ระบบจัดการฐานข้อมูลจะไม่อนุญาตให้รายการเปลี่ยนแปลงที่ขาดคุณสมบัติเข้าสู่การประมวลผล

2. การจัดลำดับการประมวลผลรายการเปลี่ยนแปลง

ลำดับการทำงานของรายการเปลี่ยนแปลงมีผลต่อสถานะภาพของข้อมูลโดยตรง ระบบจัดการฐานข้อมูลจะอนุญาตให้มีการประมวลผลเฉพาะลำดับรายการเปลี่ยนแปลงที่ถูกต้องเท่านั้น รายละเอียดอยู่ในตอนต่อไปในเรื่องการจัดลำดับการทำงานของรายการเปลี่ยนแปลง

3. การยกเลิกการประมวลผลรายการเปลี่ยนแปลง

ความเสียหายอันเนื่องมาจากสาเหตุใดสาเหตุหนึ่งกับฐานข้อมูลจะต้องถูกตรวจพบ และรายการเปลี่ยนแปลงที่ยังไม่ผ่านจุดสมบูรณ์ ณ เวลานั้นจะต้องถูกยกเลิก เพื่อให้ข้อมูลกลับมามีอยู่ในสถานะที่มั่นคงดั้งเดิม รวมถึงลำดับการเปลี่ยนแปลงอื่นที่เกี่ยวข้องกันด้วย

ตอนที่ 10.2 การประมวลผลรายการเปลี่ยนแปลงเบื้องต้น

หัวเรื่อง

- เรื่องที่ 10.2.1 แนวคิดเรื่องระบบและการประมวลผลรายการเปลี่ยนแปลง
- เรื่องที่ 10.2.2 คุณสมบัติของการประมวลผลรายการเปลี่ยนแปลงที่จำเป็น
- เรื่องที่ 10.2.3 การจัดลำดับการทำงานของรายการเปลี่ยนแปลง

แนวคิด

1. ขั้นตอนการประมวลผลรายการเปลี่ยนแปลงมีผลโดยตรงกับข้อมูลในฐานข้อมูล ดังนั้นจึงจำเป็นต้องแน่ใจว่าผลลัพธ์จากการประมวลผลจะทำให้ข้อมูลมีความถูกต้องอยู่เสมอ
2. รายการเปลี่ยนแปลงจะต้องมีคุณสมบัติครบถ้วนตามที่กำหนดเพื่อรักษาความถูกต้องของข้อมูลเอาไว้
3. การจัดลำดับการทำงานของรายการเปลี่ยนแปลงสามารถทำได้ทั้งแบบเรียงลำดับและเสมือนเรียงลำดับ โดยยึดถือความถูกต้องของผลลัพธ์เป็นหลัก

วัตถุประสงค์

หลังจากศึกษาตอนที่ 10.2 แล้ว นักศึกษาสามารถ

1. เขียนคุณสมบัติที่จำเป็นของรายการเปลี่ยนแปลงได้
2. เขียนลำดับการประมวลผลรายการเปลี่ยนแปลงที่ถูกต้องได้

เรื่องที่ 10.2.1 แนวคิดเรื่องระบบและการประมวลผลรายการเปลี่ยนแปลง

การประมวลผลรายการเปลี่ยนแปลงซึ่งเป็นกิจกรรมสำคัญที่เกิดขึ้นในระบบฐานข้อมูล ระบบจัดการฐานข้อมูลที่ดีจะต้องมีระบบจัดการรายการเปลี่ยนแปลงที่ดีด้วย ดังนั้น ในตอนนี้จะเป็นการแนะนำให้รู้จักว่าอะไรคือรายการเปลี่ยนแปลง และวิธีการประมวลผลรายการเปลี่ยนแปลงอย่างไร ในเบื้องต้น

รายการเปลี่ยนแปลงเป็นหน่วยหนึ่งของการทำงานของระบบจัดการฐานข้อมูลซึ่งมีสิทธิ์เข้าถึงและเปลี่ยนแปลงข้อมูลต่างๆได้ รายการเปลี่ยนแปลงได้รวมเอาเหตุการณ์ต่างๆ เข้าไว้เป็นรายการเปลี่ยนแปลงหนึ่งรายการ รายการเปลี่ยนแปลงในระบบฐานข้อมูลจะต้องมีคุณสมบัติที่เป็นไปตามกฎเพื่อให้ข้อมูลยังคงสภาพอยู่ได้

รายการเปลี่ยนแปลง (Transaction) คือ การทำงานของระบบจัดการฐานข้อมูลที่ทำหน้าที่ในการจัดการบางอย่าง เช่น แก้ไข เพิ่ม หรือลบข้อมูลในฐานข้อมูล รายการเปลี่ยนแปลงส่วนใหญ่จะเป็นรายการเปลี่ยนแปลงที่เกิดจากการโปรแกรมที่มีการเข้าถึงฐานข้อมูลของผู้ใช้งานหลายคน และเป็นการทำรายการเปลี่ยนแปลงแบบออนไลน์ ตัวอย่างของรายการเปลี่ยนแปลง ได้แก่ การจองหรือซื้อตั๋วเครื่องบิน การถอนเงินจาก ATM การตรวจสอบการใช้งานบัตรเครดิต การสั่งซื้อสินค้าแบบออนไลน์ การยิงจรวดมิสไซล์ การดาวน์โหลดวีดีโอคลิป เป็นต้น

การจัดการกับรายการเปลี่ยนแปลงมีความสัมพันธ์เกี่ยวเนื่องกับหลายสิ่งในระบบจัดการฐานข้อมูลอันได้แก่

ความน่าเชื่อถือของระบบ	ระบบที่ดีไม่ควรจะเสียบ่อย
ความคงอยู่ของระบบ	ระบบที่ดีจะต้องทำงานได้ตลอดเวลา
เวลาในการตอบสนอง	ระบบจะต้องสามารถตอบสนองผู้ใช้ได้ภายใน 1 ถึง 2 วินาที
ผลลัพธ์จากการทำงาน	ระบบจะต้องสามารถประมวลผลรายการเปลี่ยนแปลงได้เป็นในจำนวนหลักพันรายการต่อหนึ่งวินาที
อัตราการขยายตัว	ระบบจะต้องรองรับการขยายขนาดของข้อมูลและระดับการจัดการที่มากขึ้นเรื่อยๆ ได้
การปรับแต่งคุณลักษณะ	ระบบจะต้องมีความสามารถที่จะถูกปรับแต่งให้มีคุณสมบัติตั้งข้างต้นด้วยต้นทุนที่ต่ำได้
ความเป็นอันหนึ่งอันเดียวกัน	ระบบจะต้องไม่สร้างผลลัพธ์ที่ไม่เหมือนกัน
ความสามารถในการกระจาย	ระบบจะต้องสามารถรองรับผู้ใช้และข้อมูลได้

การประมวลผลรายการเปลี่ยนแปลงในมุมมองของผู้ใช้ ก็คือ การรับคำขอในการประมวลผลจากอุปกรณ์แสดงผล เช่น จอภาพ และนำคำขอนั้นไปปฏิบัติตาม ซึ่งแน่นอนต้องมีการเข้าถึงฐานข้อมูลด้วย จากนั้นจึงส่งผลลัพธ์ของคำขอนั้นไปแสดงยังอุปกรณ์แสดงผลต่อไป เมื่อมองจากระบบจัดการฐานข้อมูลรายการเปลี่ยนแปลงก็คืองานหนึ่งหน่วยที่จะต้องปฏิบัติตามและให้ผลลัพธ์ที่แน่นอน

ความสำคัญของรายการเปลี่ยนแปลง

- ธุรกิจตั้งแต่ขนาดกลางถึงขนาดใหญ่มีความจำเป็นในการใช้ฐานข้อมูลเพื่อเก็บข้อมูลเกี่ยวกับระบบการผลิต ดังนั้นจึงเกิดการประมวลผลรายการเปลี่ยนแปลงขึ้นจากการใช้งานฐานข้อมูลนี้
- ระบบฐานข้อมูลมีความสำคัญในตลาดของระบบคอมพิวเตอร์อย่างมาก และมีสัดส่วนของระบบในตลาดสูงมากด้วย ในปีหนึ่งๆ มีการใช้จ่ายเงินมากกว่า 500 ล้านเหรียญในการซื้อขายระบบฐานข้อมูลเพื่องานทางด้านธุรกิจ
- ระบบฐานข้อมูลทำให้การทำธุรกิจทางอินเทอร์เน็ต หรือ ที่เรียกว่า อีคอมเมิร์ซ เกิดขึ้นและเป็นไปได้

เรื่องที่ 10.2.2 คุณสมบัติของการประมวลผลรายการเปลี่ยนแปลงที่จำเป็น

เพื่อให้แน่ใจว่าบุรณภาพของข้อมูลยังคงอยู่ และเพื่อเป็นการรักษาความถูกต้องในการทำงานของรายการเปลี่ยนแปลง ระบบจัดการฐานข้อมูลจะต้องรักษา 4 คุณสมบัติหลักของรายการเปลี่ยนแปลงที่เรียกว่า ACID Properties ไว้

1. **ความเป็นอันหนึ่งอันเดียวกัน (Atomicity)** รายการเปลี่ยนแปลงจะถูกทำให้ได้ผลลัพธ์หรือไม่ถูกทำเลย จะไม่มีการทำแค่ครึ่งเดียวหรือบางส่วนของรายการเปลี่ยนแปลง

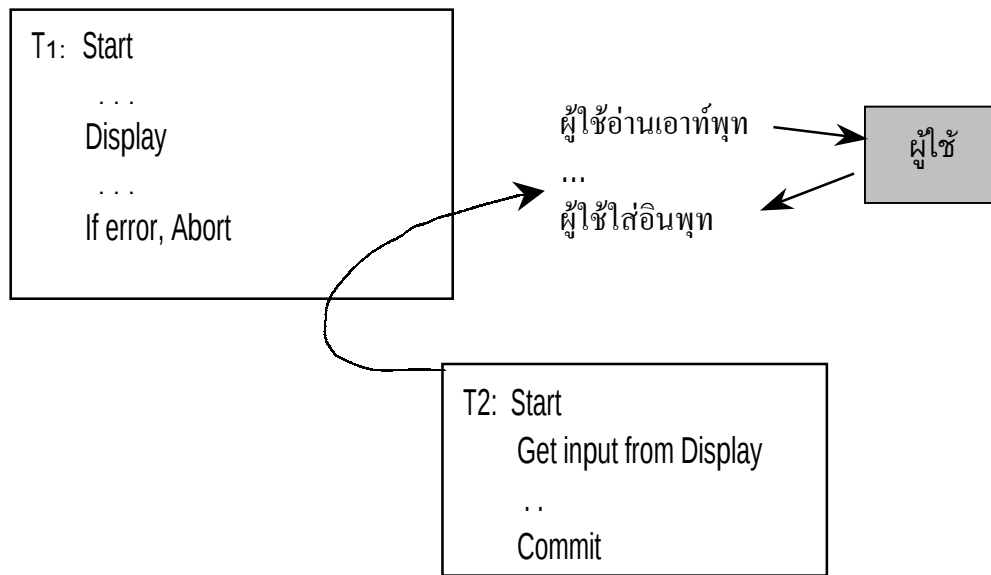
2. ความถูกต้อง (Consistency) รายการเปลี่ยนแปลงจะถูกทำแยกกัน เพื่อคงความถูกต้องของข้อมูล
3. การแยกตัว (Isolation) ถึงแม้ว่าจะมีรายการเปลี่ยนสองรายการที่พยายามทำงานพร้อมกัน ระบบจัดการฐานข้อมูลก็จะจัดการให้รายการเปลี่ยนแปลงรายการใดรายการหนึ่งได้ทำงานก่อน และเริ่มทำรายการเปลี่ยนแปลงรายการต่อไปเมื่อเสร็จสิ้นรายการเปลี่ยนแปลงก่อนหน้านี้แล้ว เพื่อรับประกันว่าจะไม่มีรายการเปลี่ยนแปลงสองรายการใดทำงานพร้อมกันอย่างแน่นอน
4. ความคงทน (Durability) ผลลัพธ์ที่เกิดจากการทำงานของรายการเปลี่ยนแปลงจะยังคงอยู่ในฐานข้อมูลถึงแม้ว่าระบบจะหยุดทำงาน

1.ความเป็นอันหนึ่งอันเดียวกัน (Atomicity)

การทำรายการเปลี่ยนแปลงแต่ละรายการจะได้ผลลัพธ์สองอย่าง คือ ได้ผลลัพธ์ตามรายการเปลี่ยนแปลงที่ระบุไว้ หรือไม่ได้ผลลัพธ์อะไรเลย ดังที่กล่าวไว้ข้างต้น เมื่อรายการเปลี่ยนแปลงถูกทำแล้วจะต้องทำให้เสร็จ หรือไม่ทำเลย ถ้าระบบเกิดล้มขึ้นมาก่อนที่จะทำรายการเปลี่ยนแปลงเสร็จแล้ว ผลลัพธ์ที่เกิดจากการทำรายการเปลี่ยนแปลงถึง ณ จุดที่ระบบเกิดความเสียหายนั้นจะไม่ถูกนำมาเขียนลงฐานข้อมูล ดังนั้นเมื่อระบบเริ่มทำงานใหม่ รายการเปลี่ยนแปลงนั้นจะถูกทำใหม่ตั้งแต่เริ่มต้น หรือ ถ้าเกิดความผิดพลาดหรือเกิดกรณีพิเศษระหว่างการทำรายการเปลี่ยนแปลงแล้ว ผลลัพธ์บางส่วนที่เกิดขึ้นจากการทำรายการเปลี่ยนแปลงก็就会被ยกเลิก และสถานะของฐานข้อมูลก็จะย้อนกลับไป ณ จุดก่อนการทำรายการเปลี่ยนแปลง ยกตัวอย่าง การโอนเงิน ประกอบด้วย การถอนเงินจากบัญชีหนึ่ง และฝากเงินเข้าอีกบัญชีหนึ่ง การทำรายการเปลี่ยนแปลงของการโอนเงินจะได้ผลลัพธ์ คือ ถอนเงินจากบัญชี และโอนเงินเข้าบัญชีเป็นที่เรียบร้อย หรือไม่ทำทั้งสองอย่าง

จุดที่รายการเปลี่ยนแปลงที่ถูกทำงานได้ผลลัพธ์เรียบร้อย เรียกว่า จุดที่ทำให้รายการเปลี่ยนแปลงสมบูรณ์ (Commit) และ รายการเปลี่ยนแปลงที่เกิดความผิดพลาดและไม่ถูกทำเลย เรียกว่า การยกเลิกรายการเปลี่ยนแปลง (Abort) ทั้งการทำให้เสร็จสมบูรณ์และการยกเลิก เรียกว่า เป็นการกระทำที่ไม่สามารถถอยกลับคืนได้ (Irrevocable action)

การยกเลิก คือ การลบล้างสิ่งที่ได้ทำไปแล้ว ในปฏิบัติการเกี่ยวกับฐานข้อมูล การยกเลิกคือการนำค่าเดิมก่อนเริ่มทำรายการเปลี่ยนแปลงกลับมาเก็บเป็นค่าปัจจุบัน แต่ในโลกของความจริง การกระทำบางอย่างก็ไม่สามารถถูกยกเลิกได้ เช่น การโอนเงิน การพิมพ์ตัว การยิงจรวด ตัวอย่างของการกระทำที่ไม่สามารถยกเลิกสิ่งที่ทำไปแล้วได้ คือ การจ่ายเงินของเครื่อง ATM ดังรูปข้างล่าง จากรูปจะพบว่า รูป (ก) สามารถยกเลิกรายการเปลี่ยนแปลงที่เกิดขึ้นหลังจุดสมบูรณ์ของรายการเปลี่ยนแปลงได้เนื่องจากรายการเปลี่ยนแปลงยังไม่ได้ถูกกระทำจนสมบูรณ์ ในรูป (ข) เมื่อระบบเกิดความเสียหายหลังจากการ “จ่ายเงิน” ซึ่งผ่านจุดสมบูรณ์ของรายการเปลี่ยนแปลงมาแล้ว จะไม่สามารถยกเลิกรายการเปลี่ยนแปลง T1 จนถึงบรรทัดของรายการ “จ่ายเงิน” ได้เนื่องจากผ่านจุดสมบูรณ์ของรายการเปลี่ยนแปลงไปแล้ว



รูปที่ 10-1 การอ่านเอาต์พุตเมื่อรายการเปลี่ยนแปลงยังไม่ถึงจุดสมบูรณ์ ไม่สามารถยกเลิกได้

2. ความถูกต้อง (Consistency)

รายการเปลี่ยนแปลงทุกรายการจะต้องรักษาความถูกต้องของฐานข้อมูลเอาไว้ โดยการรักษามูลค่าของการอ้างอิงแอททริบิวต์ (Referential Integrity) เช่น การสั่งซื้อสินค้าจะต้องใช้ข้อมูลหมายเลขลูกค้าและหมายเลขสินค้าที่มีอยู่จริงในฐานข้อมูล

การรักษาความถูกต้องของข้อมูลเป็นคุณสมบัติหนึ่งของรายการเปลี่ยนแปลงซึ่งต้องรักษาไว้ให้ได้ ดังนั้นถ้ารายการเปลี่ยนแปลงแต่ละรายการมีความถูกต้อง ลำดับของรายการเปลี่ยนแปลงก็จะต้องมีความถูกต้องเช่นเดียวกัน

T1: Start; A = Read(x); A = A - 1; Write(y,A); Commit;	T2 : Start B = Read(x); C = Read(y); If (B > C + 1) B = B - 1; Write(x,B); Commit;
--	---

จากตัวอย่างข้างบน พบว่า ความถูกต้องจะเกิดขึ้น ถ้า

$$x > y$$

ลำดับของการทำงานในรายการเปลี่ยนแปลงถูกต้อง

3. การแยกตัว (Isolation)

รายการเปลี่ยนแปลงแต่ละรายการจะถูกแยกออกจากกัน นั่นคือ ผลลัพธ์จากการทำรายการเปลี่ยนแปลงจะเป็นที่รู้จักกันก็ต่อเมื่อถึงจุดสมบูรณ์ของรายการเปลี่ยนแปลง ดังนั้นในขณะที่มีการทำรายการเปลี่ยนแปลง

แปลงรายการหนึ่ง ผลลัพธ์ของรายการเปลี่ยนแปลงนี้จะเปิดเผยแต่รายการเปลี่ยนแปลงอื่นก็ต่อเมื่อถึงจุดสมบูรณ์ของรายการนี้เท่านั้น คุณสมบัตินี้ใช้ได้กับทุกรายการเปลี่ยนแปลง

โดยธรรมชาติของรายการเปลี่ยนแปลง ผลลัพธ์จากการทำลำดับของรายการเปลี่ยนแปลงจะต้องเหมือนกันกับการทำรายการเปลี่ยนแปลงแต่ละรายการแยกกัน ถ้ามีการเปลี่ยนแปลงลำดับการทำงานของรายการเปลี่ยนแปลง (interleaved execution) ลำดับที่เปลี่ยนไปนั้นจะมีความเป็นลำดับของการทำงานที่ถูกต้องอยู่ถ้าผลลัพธ์จากการทำงานตามลำดับใหม่นั้นเหมือนกันกับผลลัพธ์ของลำดับเดิม

เมื่อผู้ใช้แต่ละคนทำรายการเปลี่ยนแปลงที่เครื่องของตนเอง จะส่งผลให้เกิดรายการเปลี่ยนแปลงจำนวนหลายรายการที่จะต้องถูกกระทำไปพร้อมกัน ซึ่งผลลัพธ์จากการทำรายการเปลี่ยนแปลงในเวลาเดียวกันนี้สามารถทำให้ผลลัพธ์ที่ไม่ถูกต้องได้ ถ้าไม่มีการรักษาลำดับของรายการเปลี่ยนแปลงไว้

4. ความคงทน (Durability)

เมื่อรายการเปลี่ยนแปลงถึงจุดสมบูรณ์ของมันแล้ว ผลลัพธ์จากการทำรายการนี้จะถูกเขียนลงจานแม่เหล็กในฐานข้อมูล และจะยังปรากฏอยู่ในฐานข้อมูล ถึงแม้ว่าหลังจากจุดสมบูรณ์จะเกิดความเสียหายกับระบบอันเนื่องมาจากโปรแกรมประยุกต์ ระบบปฏิบัติการ ระบบฐานข้อมูล หรือจานแม่เหล็ก

การเปลี่ยนแปลงที่เกิดจากการทำรายการเปลี่ยนแปลงจะถูกเขียนลงในแฟ้มข้อมูลซึ่งเก็บไว้ในจานแม่เหล็ก ข้อมูลที่บันทึกไว้นี้สามารถนำมาใช้เมื่อระบบฐานข้อมูลถูกเปิดขึ้นมาทำงานใหม่หลังจากความเสียหาย คุณสมบัติดังกล่าวทำให้ผลลัพธ์ของการทำรายการเปลี่ยนแปลงมีความน่าเชื่อถือมากขึ้น

เรื่องที่ 10.2.3 การจัดลำดับการทำงานของรายการเปลี่ยนแปลง

ระบบประมวลผลรายการเปลี่ยนแปลงอนุญาตให้รายการเปลี่ยนแปลงหลายรายการสามารถทำพร้อมกันได้ การทำรายการเปลี่ยนแปลงในเวลาเดียวกันนั้นมีผลโดยตรงกับความถูกต้องของฐานข้อมูล ดังนั้นจึงต้องมีขบวนการที่ช่วยให้รายการเปลี่ยนแปลงทำงานและเขียนข้อมูลลงในฐานข้อมูลได้อย่างถูกต้อง การจัดลำดับการทำงานของรายการเปลี่ยนแปลงเป็นวิธีที่จะช่วยรักษาลำดับการทำงานที่ถูกต้องไว้ ลำดับในการทำงานนั้นมีได้ 2 แบบ คือ

การจัดรายการเปลี่ยนแปลงแบบลำดับ (Serial schedule) วิธีนี้จะจัดรายการเปลี่ยนแปลงแบบเป็นลำดับเท่านั้น นั่นคือ รายการเปลี่ยนแปลงหนึ่งรายการจะใช้เวลาอย่างต่อเนื่องในการทำงานโดยไม่อนุญาตให้รายการเปลี่ยนแปลงใดมาแทรกเวลาในการทำงานได้

การจัดรายการเปลี่ยนแปลงแบบลำดับเสมือน (Serializable schedule) การจัดรายการเปลี่ยนแปลงแบบนี้จะอาศัยลำดับที่เทียบเท่า (Equivalent schedules) โดยวัดจากผลลัพธ์ที่เกิดขึ้น กล่าวคือ ลำดับของรายการเปลี่ยนแปลงชุดที่หนึ่ง จะถือว่าเทียบเท่ากับลำดับของรายการเปลี่ยนแปลงชุดที่สอง เมื่อผลลัพธ์ที่เกิดขึ้นจากการทำงานของลำดับทั้งสองชุดมีค่าเท่ากัน เมื่อรายการเปลี่ยนแปลงมีคุณภาพของข้อมูล ลำดับหรือ ลำดับที่เทียบเท่าของรายการเปลี่ยนแปลงจะมีคุณภาพของข้อมูลด้วย

ตอนที่ 10.3 การจัดการรายการเปลี่ยนแปลง

หัวเรื่อง

- เรื่องที่ 10.3.1 การกู้คืน
- เรื่องที่ 10.3.2 การประมวลผลรายการเปลี่ยนแปลงอย่างเป็นลำดับ
- เรื่องที่ 10.3.3 การจัดลำดับการทำงานของรายการเปลี่ยนแปลง

แนวคิด

1. ข้อมูลเป็นสิ่งที่มีความสำคัญมาก ความเสียหายที่ขึ้นกับระบบคอมพิวเตอร์อาจจะมีผลกระทบกับข้อมูลในฐานข้อมูล ดังนั้นการทำให้ข้อมูลมีความถูกต้องสมบูรณ์ดั้งเดิม จึงเป็นเรื่องที่จำเป็นมากเช่นเดียวกัน
2. เมื่อทราบสาเหตุที่ทำให้เกิดความเสียหายขึ้นกับข้อมูล ก็สามารถที่จะหาวิธีแก้ไขและกู้ข้อมูลคืนได้

วัตถุประสงค์

หลังจากศึกษาตอนที่ 1.3 แล้ว นักศึกษาสามารถ

1. เรียนรู้และเข้าใจถึงภาวะการเกิดความเสียหายของข้อมูลและวิธีการกู้คืน
2. เขียนถึงวิธีการจัดการกับความเสียหายที่เกิดขึ้นกับข้อมูลได้

เรื่องที่ 10.3.1 การกู้คืน

ระบบความสามารถในการกู้คืนจะเป็นตัวช่วยให้ระบบฐานข้อมูลสามารถกลับสู่สภาพเดิมก่อนที่ระบบจะเกิดความเสียหายได้ และจะต้องรักษาบูรณภาพของข้อมูลไว้ได้ทั้งหมด โดยรักษาสถานะของข้อมูลที่เป็นผลลัพธ์ของรายการเปลี่ยนแปลงเมื่อผ่านจุดสมบูรณ์แล้วไว้ และเพิกเฉยต่อผลลัพธ์ที่เกิดจากรายการเปลี่ยนแปลงที่ยังไม่ผ่านจุดสมบูรณ์ ตามคุณสมบัติความเป็นอันหนึ่งอันเดียวกัน (Atomicity) และความคงทน (Durability) ความเสียหายของระบบฐานข้อมูลเกิดขึ้นได้จากหลายสาเหตุ เช่น ความเสียหายอันเนื่องมาจากแหล่งเก็บข้อมูล ระบบจ่ายไฟ ซอฟต์แวร์ เป็นต้น

เมื่อมีความเกิดความเสียหายเกิดขึ้นกับฐานข้อมูล รายการเปลี่ยนแปลงที่ถูกกระทำไปแล้วและยังไม่ผ่านจุดสมบูรณ์ของรายการเปลี่ยนแปลงจะต้องถูกยกเลิก สถานะของฐานข้อมูลจะถูกกู้กลับให้เหมือนกับสถานะก่อนการทำการรายการเปลี่ยนแปลง ซึ่งยังผลให้รายการเปลี่ยนแปลงที่มีความสัมพันธ์กับรายการเปลี่ยนแปลงดังกล่าวจะต้องถูกยกเลิกด้วย ลำดับการทำงานของรายการเปลี่ยนแปลงที่มีผลกับการกู้คืน แบ่งออกเป็น 2 ประเภท คือ

1. ลำดับรายการเปลี่ยนแปลงที่สามารถกู้คืนได้ (Recoverable schedule) และ
2. ลำดับรายการเปลี่ยนแปลงที่ไม่สามารถกู้คืนได้ (Non-recoverable schedule)

ลำดับของรายการการเปลี่ยนแปลงที่ถูกสร้างขึ้นเพื่อรอการประมวลผลจะต้องเป็นลำดับที่สามารถกู้คืนภายหลังได้ ตัวอย่างลำดับรายการเปลี่ยนแปลงทั้งสองประเภท แสดงได้ดังลำดับข้างล่างนี้

รายการเปลี่ยนแปลงที่ 1	รายการเปลี่ยนแปลงที่ 2
Read(A)	
Write(A)	
Read(B)	
Commit	Read(A)
	Commit

รูปที่ 2-3 การจัดลำดับรายการเปลี่ยนแปลงแบบกู้คืนได้

รายการเปลี่ยนแปลงที่ 1	รายการเปลี่ยนแปลงที่ 2
Read(A)	
Write(A)	
	Read(A)
Read(B)	Commit
Commit	

รูปที่ 2-4 การจัดลำดับรายการเปลี่ยนแปลงแบบกู้คืนได้

จากการจัดลำดับข้างต้น เมื่อระบบเกิดความเสียหายหลังจากลำดับที่ 2 ได้ผ่านจุดสมบรูณ์ไปแล้ว ในขณะที่ลำดับที่ 1 ยังไม่ถึงจุดสมบรูณ์ เป็นผลให้ Read(A), Write(A) และ Read(B) ในลำดับที่ 1 ถูกยกเลิกทั้งหมด ในขณะที่ลำดับที่สองไม่ถูกยกเลิก เหตุการณ์นี้เป็นผลให้ฐานข้อมูลอยู่ในสถานะที่ไม่มั่นคง เนื่องจากสูญเสียบูรณภาพของข้อมูลจาก Read(A) ของลำดับที่ 2 ซึ่งเป็นผลมาจาก Write(A) ของลำดับที่ 1 ดังนั้นระบบจัดการฐานข้อมูลจะต้องจัดลำดับของรายการเปลี่ยนแปลงให้ลำดับที่ 2 ผ่านจุดสมบรูณ์หลังจากลำดับที่ 1 ผ่านจุดสมบรูณ์แล้วเสียก่อน เพื่อรักษาบูรณภาพของข้อมูลไว้

เรื่องที่ 10.3.2 การประมวลผลรายการเปลี่ยนแปลงอย่างเป็นลำดับ

ระบบจัดการฐานข้อมูลสมัยใหม่อนุญาตให้มีการทำการเปลี่ยนแปลงหลายรายการพร้อมกันได้ ดังนั้นจึงจำเป็นจะต้องมีการตรวจสอบอยู่เสมอว่าผลลัพธ์ที่เกิดจากการทำงานของรายการเปลี่ยนแปลงเหล่านั้นถูกต้อง ในการที่จะตรวจสอบความถูกต้องของข้อมูลอันเนื่องมาจากการทำงานของรายการเปลี่ยนแปลงนั้น เราสามารถตรวจสอบได้ตั้งแต่ก่อนจะเริ่มทำตามรายการเปลี่ยนแปลง นั่นคือ การตรวจสอบที่ตัวรายการเปลี่ยนแปลงเองว่ามีขั้นตอนการทำงานใดบ้าง และลำดับขั้นตอนการทำงานในรายการเปลี่ยนแปลงนั้นถูกต้องหรือไม่ ถ้าไม่ถูกต้องก็ทำการแก้ไขหรือไม่ทำการเปลี่ยนแปลงนั้น

ในการตรวจสอบการทำงานภายในรายการเปลี่ยนแปลงนั้น การทำงานที่สนใจจะมีอยู่เพียงสองรายการเท่านั้น คือ รายการอ่าน (Read) และรายการเขียน (Write) เนื่องจากเป็นรายการที่มีผลกับความถูกต้องของการทำงานโดยตรง รายการอ่านและเขียนที่สนใจขณะใดขณะหนึ่งสามารถเขียนให้อยู่ในรูป $Read(M)$ และ $Write(N)$ เมื่อ M และ N เป็นข้อมูลที่ถูกอ่านหรือเขียน การอ่านและเขียนที่เหล่านี้อาจสามารถถูกจัดลำดับใหม่ได้แต่ต้องอยู่ภายใต้เงื่อนไขต่อไปนี้

1. การจัดลำดับโดยการสลับกันอย่างสมมูล (Conflict serializability)
2. การจัดลำดับโดยใช้ผลลัพธ์เสมือน (View serializability)

การจัดลำดับโดยการสลับกันอย่างสมมูล

ระบบจัดการฐานข้อมูลสามารถจัดลำดับการทำงานของรายการเปลี่ยนแปลงใหม่ แต่ยังคงให้ผลการทำงานที่สมมูลกับลำดับเดิม โดยพิจารณาจากลำดับการการอ่านและเขียนภายใน ลำดับที่การทำงานของ S ประกอบด้วยชุดของรายการเปลี่ยนแปลง T_i และ T_j ซึ่งประกอบไปด้วยลำดับการอ่านและเขียนภายใน ระบบจัดการฐานข้อมูลจะสลับ $Operation(M)_i$ และ $Operation(N)_j$ ถ้า M และ N คือข้อมูลคนละตัวกัน และรักษาลำดับเดิมของ $Operation(M)_i$ และ $Operation(N)_j$ เอาไว้ถ้า M และ N คือข้อมูลตัวเดียวกัน การสลับลำดับการทำงานของรายการเปลี่ยนแปลงที่ใช้กระทำการบนข้อมูลตัวเดียวกันมีหลักการดังนี้

1. ถ้าชนิดของการกระทำเป็นการอ่านทั้งคู่ เช่น $Read(M)_i$ และ $Read(M)_j$ สามารถสลับได้ทันทีและจะสลับให้ $Read(M)_j$ หรือ $Read(M)_i$ เริ่มก่อนก็ได้
2. ถ้าชนิดของการกระทำเป็นการเขียนทั้งคู่ เช่น $Write(M)_i$ และ $Write(M)_j$ การกระทำทั้งสองไม่มีผลกระทบต่อกันโดยตรงแต่ละมีผลกระทบต่อการอ่านที่ตามมา ดังนั้นจึงไม่สามารถสลับได้
3. ถ้าชนิดของการกระทำเป็นการอ่านก่อนการเขียน เช่น $Read(M)_i$ และ $Write(M)_j$ แล้ว T_j จะต้องเขียนข้อมูลหลังจาก T_i อ่าน M ไปแล้วเท่านั้น และ T_j จะไม่อ่านข้อมูลที่เขียนโดย T_i
4. ถ้าชนิดของการกระทำเป็นการเขียนก่อนการอ่าน เช่น $Write(M)_i$ และ $Read(M)_j$ แล้ว T_j จะต้องอ่านข้อมูลหลังจาก T_i เขียน M ไปแล้วเท่านั้น และ T_j จะต้องเขียนข้อมูลก่อนที่ T_i จะอ่านค่า M ไป

การกระทำการบนข้อมูลตัวเดียวกันจากรายการเปลี่ยนแปลงคนละชุด และมีการกระทำอย่างน้อยหนึ่งอย่างเป็นการอ่าน เรียกว่า Conflict Operation การกระทำการบนข้อมูลคนละตัวกันไม่เรียกว่าเป็น Conflict Operation ดังนั้นในการสลับตำแหน่งการทำงานกับข้อมูลจะพิจารณาจาก Non-Conflict Operation นั่นคือการกระทำการบนข้อมูลคนละตัวกัน หรือ การอ่านข้อมูลตัวเดียวกัน สามารถสลับตำแหน่งการกระทำได้ ลำดับใหม่ที่ได้การสลับลำดับ S โดยพิจารณาจาก Non-Conflict Operation ในที่นี้จะเรียกว่าลำดับ S เสมือน หรือ Conflict Serializability

ตัวอย่างของลำดับที่เป็น Conflict Serializability แสดงได้ดังลำดับต่อไปนี้

รายการเปลี่ยนแปลงที่ 1	รายการเปลี่ยนแปลงที่ 2
Read(A)	
Write(A)	
	Read(A)
	Write(A)
Read(B)	
Write(B)	
	Read(B)
	Write(B)

ลำดับที่ 1

ลำดับที่ 2

รายการเปลี่ยนแปลงที่ 1	รายการเปลี่ยนแปลงที่ 2
Read(A)	
Write(A)	
	Read(A)
Read(B)	
	Write(A)
Write(B)	
	Read(B)
	Write(B)

ลำดับที่ 2 คือลำดับที่เกิดจากการสลับตำแหน่งของการกระทำในลำดับที่ 1 โดยการสลับ Write(A) ของรายการเปลี่ยนแปลงที่ 2 และ Read(B) ของรายการเปลี่ยนแปลงที่ 1 จากลำดับที่ 2 จะเห็นว่ามีการกระทำอีกหลายคู่ที่เป็น Non-Conflict Operation และสามารถสลับกันได้ เช่น

Read(A) ของรายการเปลี่ยนแปลงที่ 2 และ Read(B) ของรายการเปลี่ยนแปลงที่ 1 เมื่อสลับแล้วจะได้ลำดับที่ 3

จากลำดับที่ 3 สลับ Write(A) ของรายการเปลี่ยนแปลงที่ 2 และ Write(B) ของรายการเปลี่ยนแปลงที่ 1 เมื่อสลับแล้วจะได้ลำดับที่ 4

จากลำดับที่ 4 สลับ Read(A) ของรายการเปลี่ยนแปลงที่ 2 และ Write(B) ของรายการเปลี่ยนแปลงที่ 1 เมื่อสลับแล้วจะได้ลำดับที่ 5

ลำดับที่ 3

รายการเปลี่ยนแปลงที่ 1	รายการเปลี่ยนแปลงที่ 2
Read(A)	
Write(A)	
Read(B)	
	Read(A)
	Write(A)
Write(B)	
	Read(B)
	Write(B)

ลำดับที่ 4

รายการเปลี่ยนแปลงที่ 1	รายการเปลี่ยนแปลงที่ 2
Read(A)	
Write(A)	
Read(B)	
	Read(A)
Write(B)	
	Write(A)
	Read(B)
	Write(B)

ลำดับที่ 5

รายการเปลี่ยนแปลงที่ 1	รายการเปลี่ยนแปลงที่ 2
Read(A)	
Write(A)	
Read(B)	
Write(B)	
	Read(A)
	Write(A)
	Read(B)
	Write(B)

การจัดลำดับโดนใช้ผลลัพธ์เสมือน

รายการเปลี่ยนแปลงบางชุดไม่สามารถจัดลำดับโดยใช้การสลับกันได้เนื่องจากตำแหน่งของการกระทำเป็นเรื่องสำคัญ ในรายการเปลี่ยนแปลงแบบนี้จะยึดถือที่ผลลัพธ์จากการทำรายการเปลี่ยนแปลงเป็นสำคัญ ดังนั้นลำดับ S จะมีลำดับ S เสมือนเมื่อให้ผลลัพธ์การทำงานที่เหมือนกัน (View Serializability) ใน การพิจารณาลำดับการทำงานสองลำดับว่าให้ผลลัพธ์ที่เหมือนกันหรือไม่ สามารถพิจารณาได้จากเงื่อนไข 3 ข้อด้วยกัน คือ

เรื่องที่ 10.3.3 การจัดการรายการเปลี่ยนแปลงใน SQL

ภาษาในการจัดการฐานข้อมูล ดังเช่น SQL มีโครงสร้างที่ใช้ในการจัดการกับรายการเปลี่ยนแปลงเพื่ออำนวยความสะดวกให้กับโปรแกรมเมอร์ โดยที่โปรแกรมเมอร์จะต้องกำหนดลำดับการทำรายการเปลี่ยนแปลงซึ่งจะถูกนำไปปฏิบัติโดยระบบจัดการฐานข้อมูลที่ไม่มีความเสี่ยงต่อความมั่นคงของข้อมูล

รายการเปลี่ยนแปลงใน SQL จะต้องสามารถจัดลำดับการทำงานเสมือนได้ และสามารถกู้คืนได้เมื่อเกิดความผิดพลาดใดๆ ขึ้น คำสั่งในภาษา SQL ที่ใช้ในการจัดการกับรายการเปลี่ยนแปลงโดยตรงได้แก่คำสั่งที่ใช้ออกจุดสิ้นสุดของรายการเปลี่ยนแปลง คือ COMMIT และ ROLLBACK ทั้งสองคำสั่งมีผลต่อการสิ้นสุดรายการเปลี่ยนแปลงที่แตกต่างกัน กล่าวได้ดังนี้

- COMMIT [work] ใช้ในการยืนยันผลการเปลี่ยนแปลงที่เกิดขึ้นกับฐานข้อมูล เพื่อให้ผลลัพธ์นั้นคงอยู่ตลอดไปจนกว่าจะมีคำสั่งในการเปลี่ยนแปลงข้อมูล
- ROLLBACK [work] ใช้ในการยกเลิกการเปลี่ยนแปลงที่มีต่อฐานข้อมูลจากการทำรายการเปลี่ยนแปลงชุดปัจจุบันจนถึงจุดสุดท้ายที่ใช้คำสั่ง COMMIT

SQL ไม่มีคำสั่งที่ใช้ในการกำหนดจุดเริ่มต้นของรายการเปลี่ยนแปลงโดยตรง การเริ่มต้นทำรายการเปลี่ยนแปลงจะถูกทำโดยอ้อมโดยการทำคำสั่ง SQL ใดๆ แต่ทุกครั้งที่มีการทำรายการเปลี่ยนแปลงจะต้องมีการใช้คำสั่งในการกำหนดจุดสิ้นสุดของรายการเปลี่ยนแปลงดังเช่นสองคำสั่งที่กล่าวข้างต้น นอกจากนี้ SQL ยังมีคำสั่งที่ใช้ในการกำหนดคุณลักษณะของรายการเปลี่ยนแปลง เช่น สิทธิการเข้าถึง และระดับของความมั่นคง

คำสั่งในภาษา SQL สามารถแบ่งออกได้เป็นหลายประเภท ได้แก่

1. คำสั่งที่ใช้กำหนดค่าให้กับข้อมูล (Data Definition Language commands)
2. คำสั่งที่ใช้จัดการกับข้อมูล (Data Manipulation Language commands)
3. คำสั่งที่ใช้ในการควบคุมรายการเปลี่ยนแปลง (Transaction Control commands)
4. คำสั่งที่ใช้ในการควบคุมช่วงการทำงาน (Session Control commands)
5. คำสั่ง COMMIT และ ROLLBACK เป็นคำสั่งที่อยู่ในประเภทที่ใช้ในการควบคุมรายการเปลี่ยนแปลง

ดังที่ได้กล่าวข้างต้นว่าการทำการเปลี่ยนแปลงใน SQL สามารถกำหนดคุณลักษณะของรายการเปลี่ยนแปลงได้ คุณลักษณะหนึ่งก็คือ สิทธิของการเข้าถึง อันได้แก่ อ่านอย่างเดียว (read only) หรือ อ่านและเขียน (read write) ถ้าไม่กำหนดสิทธิที่จะถูกใช้ก็คือ อ่านและเขียน

SQL-92 ได้กำหนดไว้ว่ารายการเปลี่ยนแปลงสามารถถูกกระทำได้ถึงแม้ว่าลำดับของรายการเปลี่ยนแปลงจะไม่ใช้ลำดับเสมือน (nonserializable) เช่น การทำการเปลี่ยนแปลงในระดับ read uncommitted ซึ่งอนุญาตให้รายการเปลี่ยนแปลงสามารถอ่านข้อมูลที่ยังไม่ถูก commit ได้ การกำหนดคุณลักษณะนี้จะสามารถทำได้เมื่อรายการเปลี่ยนแปลงนั้นสามารถทำได้โดยใช้ข้อมูลโดยประมาณที่ไม่ต้องการความถูกต้องแม่นยำสูงมากนัก

ระดับของความมั่นคงที่สามารถกำหนดได้ใน SQL-92 ได้แก่

- **Serializable** เมื่อไม่กำหนดคุณลักษณะนี้จะถูกใช้กับทุกรายการเปลี่ยนแปลง
- **Repeatable read** คุณลักษณะนี้จะใช้ได้เฉพาะกับข้อมูลที่ผ่านมาการ commit แล้วเท่านั้น และระหว่างการอ่านสองครั้งจะไม่อนุญาตให้มีการเปลี่ยนแปลงข้อมูลนี้จากรายการเปลี่ยนแปลงอื่น
- **Read committed** คุณลักษณะนี้จะอนุญาตให้มีการอ่านข้อมูลที่ผ่านมาการ Commit แล้ว และไม่อนุญาตให้มีการอ่านซ้ำ ดังนั้นเมื่อมีการอ่านข้อมูลไปแล้วหนึ่งครั้ง รายการเปลี่ยนแปลงอื่นที่ผ่านการ Commit แล้วจะสามารถเปลี่ยนแปลงข้อมูลนี้ได้
- **Read uncommitted** อนุญาตให้อ่านข้อมูลที่ไม่ได้ผ่านการ commit ได้

ตอนที่ 10.4 การจัดการรายการเปลี่ยนแปลงขั้นสูง

หัวเรื่อง

- เรื่องที่ 10.4.1 การตรวจสอบการประมวลผลรายการเปลี่ยนแปลง
- เรื่องที่ 10.4.2 ระบบการประมวลผลรายการเปลี่ยนแปลงแบบทันเวลา

แนวคิด

1. การตรวจสอบความถูกต้องในการประมวลผลรายการเปลี่ยนแปลงจะช่วยรักษาความถูกต้องในการทำงานของระบบจัดการฐานข้อมูล และช่วยรักษาความถูกต้องของข้อมูลภายในฐานข้อมูลเอาไว้ได้
2. การประมวลผลข้อมูลมีหลายประเภท ข้อมูลที่ระยะเวลาในการปรากฏของข้อมูลมีความสำคัญมาก วิธีการประมวลผลที่เลือกใช้ก็ต้องเป็นแบบที่สามารถให้ผลลัพธ์ของการประมวลผลได้ในเวลาที่ต้องการ

วัตถุประสงค์

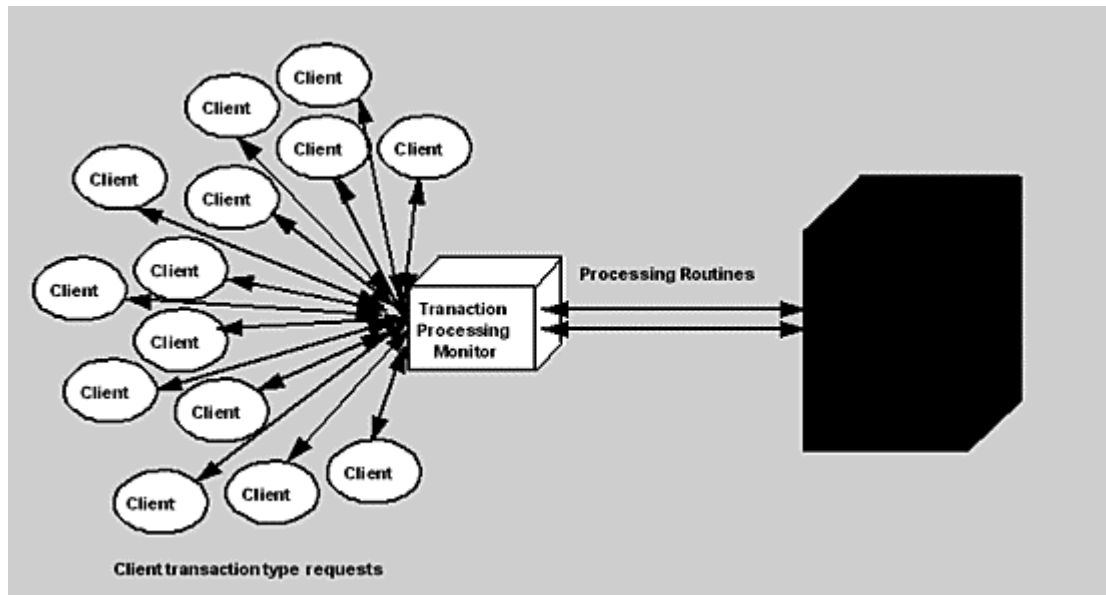
หลังจากที่ศึกษาตอนที่ 10.1 แล้ว นักศึกษาสามารถ

1. เรียนรู้วิธีการตรวจสอบความถูกต้องของการประมวลผลรายการเปลี่ยนแปลง
2. ทราบถึงระบบประมวลผลรายการเปลี่ยนแปลงขั้นสูง

เรื่องที่ 10.4.1 การตรวจสอบการประมวลผลรายการเปลี่ยนแปลง

ระบบควบคุมการทำงานของรายการเปลี่ยนแปลงได้ถูกคิดค้นขึ้นในปี 1970 เพื่อใช้ในการควบคุมการทำงานของเทอร์มินัลจำนวนมาก ในปัจจุบันระบบควบคุมนี้ได้ถูกนำมาใช้กับฐานข้อมูลเพื่อควบคุมการทำงานของรายการเปลี่ยนแปลงและปรับปรุงฐานข้อมูล ระบบควบคุมจะทำหน้าที่เป็นตัวกลางที่คอยบริการลูกข่ายในระบบแม่ข่ายลูกข่าย โดยการรับคำร้องขอจำนวนมากมาจากลูกข่ายและควบคุมการส่งคำร้องเหล่านี้ไปให้ผู้ให้บริการหรือแม่ข่าย ดังรูปที่ 10-2 โปรแกรมประยุกต์ในปัจจุบันจะมีลักษณะการทำงานเป็นแบบแม่ข่ายและลูกข่ายซึ่งจะต้องมีการติดต่อกับฐานข้อมูลจากหลายแหล่งอยู่ตลอดเวลา และยังต้องติดต่อกับแม่ข่ายในระบบอื่น เช่น ระบบไฟล์ ระบบสำรองข้อมูล เป็นต้น นอกจากนี้ยังต้องติดต่อกับโปรแกรมประยุกต์ที่อยู่บนเครื่องที่อยู่ห่างไกลออกไป จะเห็นได้ว่าโปรแกรมประยุกต์ในปัจจุบันจะต้องทำหน้าที่หลายอย่างในการจัดการข้อมูลและติดต่อกับระบบอื่นๆ และในขณะเดียวกันจะต้องรักษาคุณสมบัติ ACID ในการทำรายการเปลี่ยนแปลงผ่านระบบเหล่านี้ด้วย

ระบบควบคุมการทำงานของรายการเปลี่ยนแปลงในปัจจุบันสามารถรองรับการจัดการการทำงานกับโปรแกรมประยุกต์ดังที่กล่าวมาได้ โดยมองระบบย่อยแต่ละระบบที่จะต้องติดต่อกันเป็นแหล่งข้อมูลอันหนึ่งในการติดต่อกับแหล่งข้อมูลย่อยเหล่านี้ระบบควบคุมจะใช้ลักษณะเดียวกับการทำรายการเปลี่ยนแปลงทั่วไป คือ กำหนดจุดเริ่มต้น จุดสิ้นสุดของรายการเปลี่ยนแปลง การยกเลิกรายการเปลี่ยนแปลง เป็นต้น



รูป 10-2 โครงสร้างของระบบควบคุมการทำงานของรายการเปลี่ยนแปลง
(Transaction processing Monitor System Architecture)

นอกจากนี้ระบบควบคุมยังสามารถจัดรายการเปลี่ยนแปลงให้เหมาะสมกับบริการแต่ละประเภทและฐานข้อมูลได้ เช่นเมื่อมีความต้องการในการปรับปรุงฐานข้อมูล ระบบควบคุมจะส่งข้อความเพื่อระบุผลลัพธ์ในการทำงานและรับคำร้องขอบริการใหม่ เมื่อมีความผิดพลาดเกิดขึ้นในระบบ ผลลัพธ์ในการทำรายการเปลี่ยนแปลงหรือบริการใดๆ จะมีความเป็นอันหนึ่งอันเดียวกัน (Atomicity) คือ ถูกกระทำและยืนยันผลลัพธ์ทั้งหมด หรือยกเลิกทั้งหมด

ในระบบแม่ข่ายลูกข่าย ลูกข่ายมักจะติดต่อกับแม่ข่ายโดยผ่าน Remote Procedure Call (RPC) ที่ช่วยให้การติดต่อกับระหว่างระบบย่อยเป็นไปด้วยดีและทันเวลา ขั้นตอนการทำงานจะเริ่มที่ลูกข่ายเป็นตัวเรียกการทำงานของ RPC และแม่ข่ายเป็นตัวกระทำตาม RPC จากนั้นผลลัพธ์จึงถูกส่งกลับไปที่ลูกข่าย ระบบควบคุมอย่างเช่น Enica มีระบบการติดต่อที่ใช้ RPC ในการจัดการรายการเปลี่ยนแปลงเมื่อได้รับคำร้องจากลูกข่าย

ระบบฐานข้อมูลในปัจจุบันสนับสนุนการปรับปรุงความถูกต้องของข้อมูลโดยเขียนผลลัพธ์ของการปรับปรุง ณ ฐานข้อมูลหลักลงในฐานข้อมูลสำรองที่อยู่ ณ ตำแหน่งอื่นในเครือข่ายด้วย ดังนั้นเมื่อฐานข้อมูลอันใดอันหนึ่งล้ม ฐานข้อมูลที่เหลืออยู่ก็จะขึ้นมาทำงานเป็นฐานข้อมูลหลักแทน และระบบควบคุมก็จะทำงานเสมือนว่าไม่ได้เกิดความเสียหายขึ้นกับฐานข้อมูล เนื่องจากคำร้องขอทำรายการเปลี่ยนแปลงจะถูกส่งมาให้ระบบควบคุมก่อน และระบบควบคุมก็จะเป็นตัวจัดการส่งคำร้องนั้นไปให้ฐานข้อมูลที่ยังทำงานได้ต่อไป

เรื่องที่ 10.4.2 ระบบการประมวลผลรายการเปลี่ยนแปลงแบบทันเวลา

ในการทำงานของโปรแกรมประยุกต์บางอัน เวลาในการทำงานเป็นสิ่งที่สำคัญสิ่ง ดังนั้นจึงต้องมีการกำหนดเวลาในการทำงานให้เสร็จสิ้น ระบบที่มีเวลาเข้ามาเกี่ยวข้องนี้เรียกว่า ระบบการทำงานแบบทันเวลา (real-time systems) ความถูกต้องของผลลัพธ์ไม่ใช่เรื่องเดียวที่ควรจะให้มีความสำคัญ เรื่องที่สำคัญที่สุดก็คือ การทำงานให้เสร็จตามเวลา กำหนดเวลาในการทำงานสามารถแยกได้เป็น 2 ประเภท คือ

Hard ค่าของงานจะถูกนับเป็น 0 ถ้าไม่สามารถทำให้เสร็จภายในกำหนดเวลา

Soft ค่าของงานจะมีค่าใกล้เคียง 0 และระดับความล่าช้าจะเพิ่มขึ้น

การจัดการรายการเปลี่ยนแปลงในระบบการทำงานแบบทันเวลานี้ จะต้องพิจารณาถึงเรื่องของการกำหนดเวลาเป็นอันดับแรก ถ้ามีข้อตกลงว่าในการทำงานร่วมกันของรายการเปลี่ยนแปลง รายการเปลี่ยนแปลง Ti จะต้องรอไปก่อน เช่นนี้ Ti อาจจะไม่เสร็จทันเวลาที่กำหนด ดังนั้นจึงต้องอนุญาตให้ Ti สามารถเรียกร้องเวลาการทำงาน และหยุดการทำงานปัจจุบันก่อนเพื่อให้ Ti ได้ทำเสร็จทันเวลา อย่างไรก็ตามการหยุดรายการเปลี่ยนแปลงใดจะต้องกระทำด้วยความระมัดระวัง เนื่องจากรายการเปลี่ยนแปลงที่ถูกหยุดอาจ จะไม่สามารถถูกทำให้เสร็จทันเวลาด้วยเช่นกัน

ความลำบากในการกำหนดลำดับการทำงานของรายการเปลี่ยนแปลงในระบบการทำงานแบบทันเวลานี้มีผลมาจากเวลาที่แต่ละรายการเปลี่ยนแปลงจะต้องใช้ในการทำงานซึ่งมีความหลากหลายมาก แหล่งเก็บข้อมูลที่ใช้ในการทำรายการเปลี่ยนแปลงมีผลเป็นอย่างมากต่อเวลาที่ใช้ในการทำงาน ถ้าข้อมูลที่ต้องการทั้งหมดอยู่ใน buffer จะไม่ทำให้เกิดปัญหาในเรื่องของเวลาขึ้น แต่ถ้าต้องมีการอ่านหรือเขียนข้อมูลกับดิสก์ การประมาณเวลาที่ต้องใช้ในการอ่านหรือเขียนข้อมูลจากดิสก์จะทำได้ยาก ดังนั้นในการออกแบบระบบการทำงานแบบทันเวลาจะต้องมั่นใจได้ว่ามีองค์ประกอบพร้อมที่จะทำงานได้ในเวลาที่กำหนด