Database System

Normalization

Why do we care about design ?

- จุดประสงค์ของการออกแบบฐานข้อมูล
  - Data Integrity
  - Easy/Efficient Data Access
- จะเกิดอะไรขึ้นกับการออกแบบที่ไม่ดีพอ
  - ข้อมูลมีความซ้ำซ้อน
    - Data Redundancy
  - ไม่สามารถควบคุมข้อบังคับความถูกต้องของข้อมูลได้ (Integrity Constraint)

Example Data

Registration (Relation R)						
Course	Unit	Term	Instructor	Student	Advisor	Office
642	3	2556-1	John	Bell	Miller	521
642	3	2556-1	John	Kent	Ullman	521
642	3	2556-2	David	Brian	White	523
644	3	2556-1	David	Howard	Ullman	521
644	3	2556-2	John	Bell	Miller	521
644	3	2556-2	John	Jordan	Smith	523
700	1	2556-1	Smith	Walker	West	522
700	1	2556-2	Smith	Jordan	Smith	523
700	1	2556-2	Smith	Howard	Ullman	521
799	2	2556-1	West	Walker	West	522
799	2	2556-2	White	Bell	Miller	521

Example of Bad Design

Registration (Relation R)						
Course	Unit	Term	Instructor	Student	Advisor	Office
642	3	2556-1	John	Bell	Miller	521
642	3	2556-1	John	Kent	Ullman	521
642	3	2556-2	David	Brian	White	523
644	3	2556-1	David	Howard	Ullman	521
644	3	2556-2	John	Bell	Miller	521
644	3	2556-2	John	Jordan	Smith	523
700	1	2556-1	Smith	Walker	West	522
700	1	2556-2	Smith	Jordan	Smith	523
700	1	2556-2	Smith	Howard	Ullman	521
799	2	2556-1	West	Walker	West	522
799	2	2556-2	White	Bell	Miller	521

Key : course + Term + Student

A lot of data redundancy

Update Anomaly

Registration (Relation R)						
Course	Unit	Term	Instructor	Student	Advisor	Office
642	3	2556-1	John	Bell	Miller	521
642	3	2556-1	John	Kent	Ullman	521
642	3	2556-2	David	Brian	White	523
644	3	2556-1	David	Howard	Ullman	521
644	3	2556-2	John	Bell	Miller	521
644	3	2556-2	John	Jordan	Smith	523
700	1	2556-1	Smith	Walker	West	522
700	1	2556-2	Smith	Jordan	Smith	523
700	1	2556-2	Smith	Howard	Ullman	521
799	2	2556-2	West	Walker	West	522
799	2	2556-2	White	Bell	Miller	521

- อ. Ann จะสอนวิชา 644 ในเทอม 2556-2 แทน อ.John
- มีจำนวนเรคอร์ดเท่าใดที่จะต้องปรับปรุงข้อมูล ?

Insert Anomaly

Registration (Relation R)						
Course	Unit	Term	Instructor	Student	Advisor	Office
…	…	…	…	…	…	…
799	2	2556-2	West	Walker	West	522
799	2	2556-2	White	Bell	Miller	521
701	**2**	**2556-2**	**Ullman**	??	??	??

- อ. Ullman ต้องการเปิด Course ใหม่ 701 ใน Term 2556-2 จะไม่สามารถทำได้
- ยังไม่มีนักศึกษาที่ลงทะเบียนเรียนในวิชานี้ ดังนั้นจะไม่สามารถบันทึกข้อมูลได้ (ไม่อนุญาตให้มีค่า null ใน student [key] attribute)

Delete Anomaly

Registration (Relation R)						
Course	Unit	Term	Instructor	Student	Advisor	Office
642	3	2556-1	John	Bell	Miller	521
642	3	2556-1	John	Kent	Ullman	521
642	3	2556-2	David	Brian	White	523
644	3	2556-1	David	Howard	Ullman	521
644	3	2556-2	John	Bell	Miller	521
644	3	2556-2	John	Jordan	Smith	523
700	1	2556-1	Smith	Walker	West	522
700	1	2556-2	Smith	Jordan	Smith	523
700	1	2556-2	Smith	Howard	Ullman	521
799	2	2556-1	West	Walker	West	522
799	2	2556-2	White	Bell	Miller	521

- อ. David จะปิดวิชา 642 ในเทอม 2556-2
- แล้วจะต้องทำอย่างไรกับข้อมูลของ Brian ?

Problem in Relational Database Design

- Data Redundancy
  - การซ้ำซ้อนของข้อมูลทำให้ต้องใช้พื้นที่การเก็บข้อมูลไปโดยเปล่าประโยชน์
- Update anomaly
  - การปรับปรุงข้อมูลหากปรับปรุงไม่ครบหมดจะเกิดความไม่สอดคล้องขึ้น (Inconsistency)
- Insert anomaly
  - ในการแทรกข้อมูลอาจเกิดข้อผิดพลาดเนื่องจากข้อมูลที่เป็นไปไม่ได้ (key ไม่ครบ)
- Delete anomaly
  - การลบข้อมูบางเรคอร์ด อาจทำให้ข้อมูลที่ไม่ได้ตั้งใจลบสูญหายไปด้วย

Problem in Relational Database Design

- Solution : การแยกตาราง (Decomposition)
- เพื่อลดข้อมูลซ้ำซ้อน

Good Decomposition

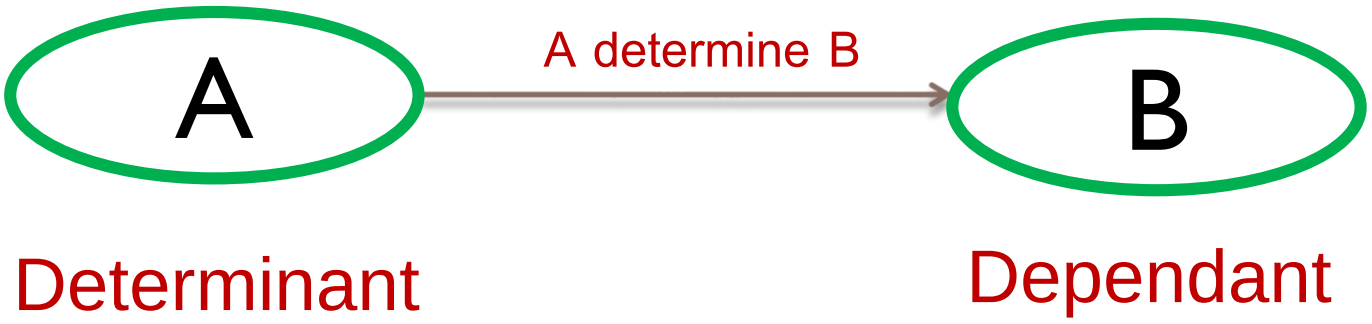
- Lossless-join Decomposition
  - หากทำการแยกตารางออกจากกันแล้ว (decomposition) เมื่อรวมกัน (join) กลับคืนมาจะสามารถทำให้ข้อมูลกลับมาสู่โครงสร้างเดิมได้
- Dependency Preservation
  - ทุกเงื่อนไขหรือข้อกำหนดเดิม (Constraint) จะต้องอยู่ครบถ้วนหลังจากการแยกตารางออกจากกัน (decomposition)

Functional Dependency (FD)

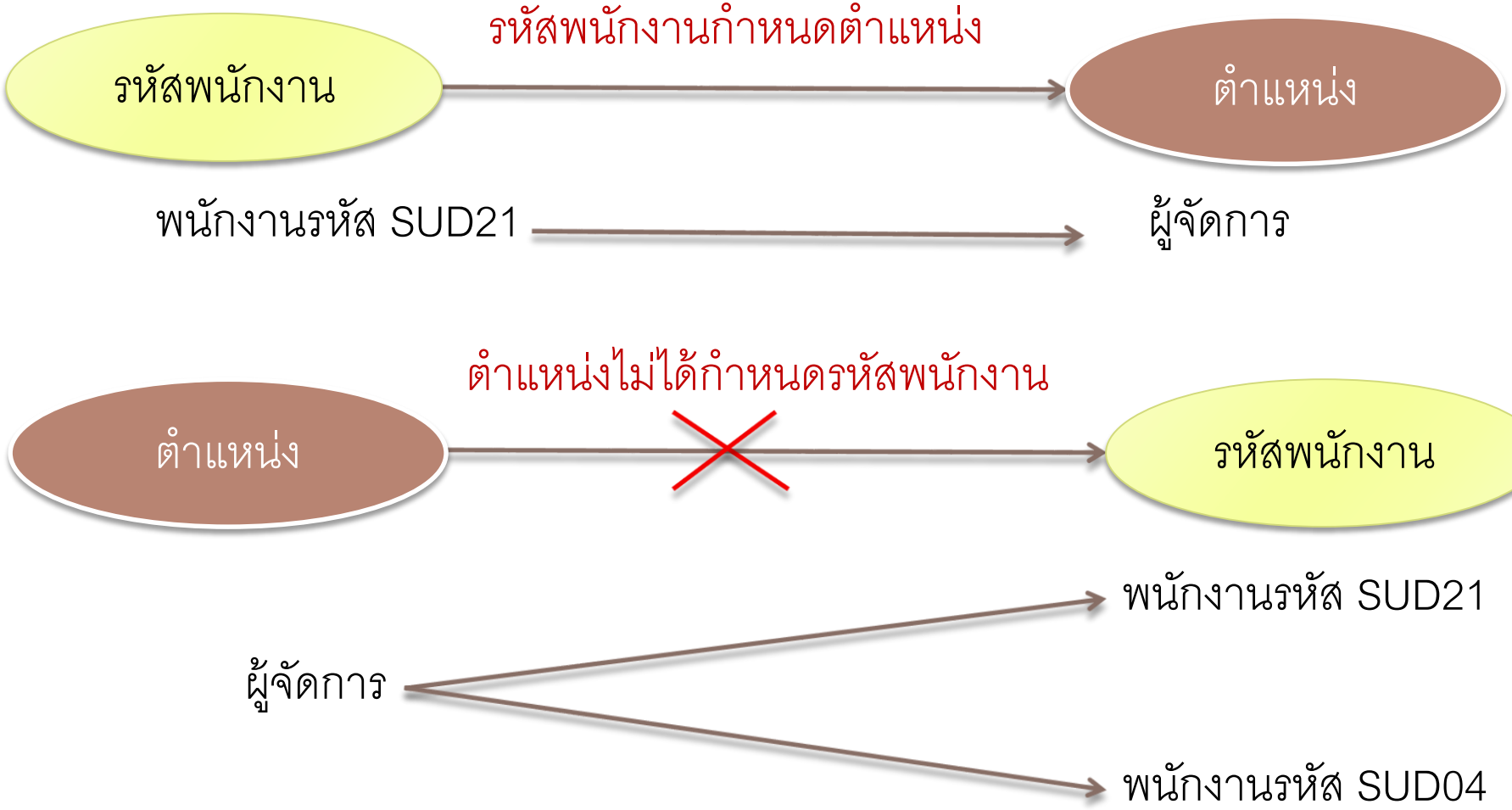
- FD คือ ความสัมพันธ์เชิงฟังก์ชั่นระหว่างแอตทริบิวต์ โดยที่แอตทริบิวต์หนึ่งหรือกลุ่มของแอตทริบิวต์ที่ประกอบกันแล้ว สามารถระบุค่าของแอตทริบิวต์อื่น ๆ ในทูเพิลเดียวกันของรีเลชันนั้นได้
- FD ได้มาจาก
  - business rules (กฎทางธุรกิจ) / Policy (นโยบาย)
  - Functional Dependency อื่น ๆ
- โดยแอตทริบิวต์ที่เป็นตัวระบุค่าในแอตทริบิวต์อื่น ๆ เรียกว่า “determinant”
- ส่วนแอตทริบิวต์อื่น ๆ ที่ถูกระบุค่าโดย determinant เรียกว่า dependent

Functional Dependency (ต่อ)

- สมมติ A และ B เป็นแอตทริบิวต์ในรีเลชัน R และ B มีความสัมพันธ์เชิงฟังก์ชันกับ A (สัญลักษณ์ A → B)
- กล่าวได้ว่า แต่ละค่าของ A สามารถระบุค่า B ได้เพียง 1 ค่า



Functional Dependency (FD) (ต่อ)



Business Rules/Policies → FD

- Business Rules/Policies
  - ทุกวิชา (course) จะประกอบด้วยจำนวนหน่วยกิต (unit)
  - นักศึกษา (student) ทุกคนมีอาจารย์ที่ปรึกษา (advisor) แค่คนเดียว
    - และอาจารย์หนึ่งคนสามารถเป็นที่ปรึกษาให้นักศึกษาได้หลายคน
  - Advisor ทุกคนมีห้องทำงาน (office) หนึ่งห้อง
    - แต่ห้องทำงานห้องหนึ่ง (office) มีอาจารย์ใช้ร่วมกัน
  - ในแต่ละเทอม (term) วิชาหนึ่งๆ (course) จะสอนโดยผู้สอน (instructor) เพียงคนเดียว
    - Student ทุกคนจะเรียนได้หลาย course ในแต่ละ term
    - Instructor ทุกคนจะสอนหลาย course ในแต่ละ term

Business Rules/Policies → FD (cont.)

- Function Dependency
  - Registration(Course, Unit, Term, Instructor, Student, Advisor, Office)
  - Course ⟶ Unit
  - Student ⟶ Advisor
  - Advisor ⟶ Office
  - Course, Term → Instructor

Relation vs. Functional Dependency

Registration (Relation R)						
Course	Unit	Term	Instructor	Student	Advisor	Office
642	3	2556-1	John	Bell	Miller	521
642	3	2556-1	John	Kent	Ullman	521
642	3	2556-2	David	Brian	White	523
644	3	2556-1	David	Howard	Ullman	521
644	3	2556-2	John	Bell	Miller	521
644	3	2556-2	John	Jordan	Smith	523
700	1	2556-1	Smith	Walker	West	522
700	1	2556-2	Smith	Jordan	Smith	523
700	1	2556-2	Smith	Howard	Ullman	521
799	2	2556-1	West	Walker	West	522
799	2	2556-2	White	Bell	Miller	521

FD สามารถใช้เพื่อ

- หา FD อื่น ๆ
- หา candidate key, primary key
- ตัวอย่าง

  $C \longrightarrow U$

  $C, T \longrightarrow I$

  $S \longrightarrow A$

  $A \longrightarrow O$
- Conclusion

  Key (SCT)

  SCT → CUTISAO

Type of Dependencies in a Relation

- Trivial Dependency
  - ถ้า B เป็น subset ของ A ดังนั้น A → B
  - ตย. SA → A
- Partial Dependency บางแอททริบิวต์ขึ้นกับ บางส่วนของคีย์หลัก
  - ถ้า AB → ABCD และ B → C ดังนั้น B → C เป็น Partial Dependency
  - ตย. SCT → CUTISAO; SCT = primary key
    - ดังนั้น C → U; CT → I; และ S → A เป็น partial dependency
- Transitive Dependency การขึ้นต่อกันแบบถ่ายทอด
  - ถ้า A → B และ B → C ดังนั้น A → C
  - ต.ย. S → A → O
    - ดังนั้น S → O เป็น Transitive Dependency

Example : R เป็น FD หรือไม่ ?

A	B	C	D	E
a1	b1	c1	d1	e1
a2	b1	c1	d1	e1
a3	b2	c1	d2	e2
a3	b2	c2	d2	e2
a4	b1	c2	d3	e1

R(A,B,C,D,E)

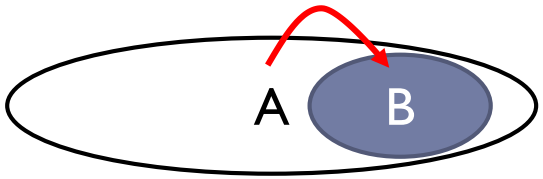
- satisfy ...
  - A → DE
  - D → BE
  - BC → DE
  - AC → B
  - AB → E
  - ...
- does not satisfy ...
  - AB → C
  - AD → C
  - CD → A
  - ...
- What might be a primary key?
  - AC

Closure of a Set of Functional Dependencies

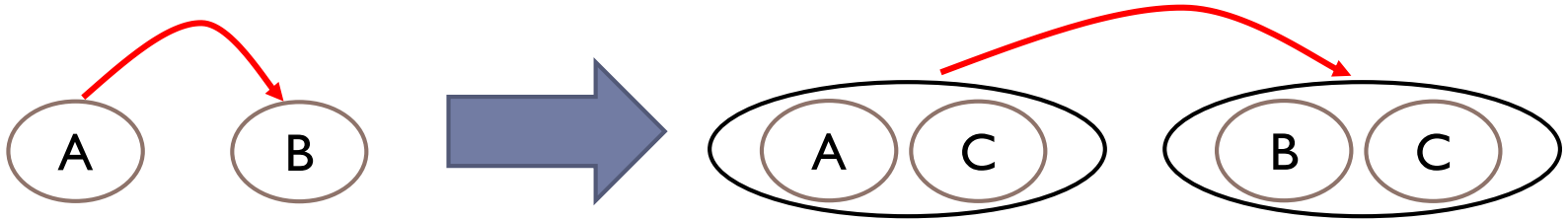
- การครอบคลุม F (F^+)
  - ให้ F เป็นเซตของ FDs
  - F+ คือ ทุกเซตที่เป็นไปได้ของ FD ที่เกิดจาก F (โดยใช้กฎของ Armstrong's Axiom)
- Armstrong's Axiom วิธีหา closure
  - กฎที่ใช้เพื่อหา F^+ จาก F
  - สอดคล้องและถูกต้องกับ F
  - ครอบคลุมสมบูรณ์ทุกกรณีของ F

Armstrong's Axioms

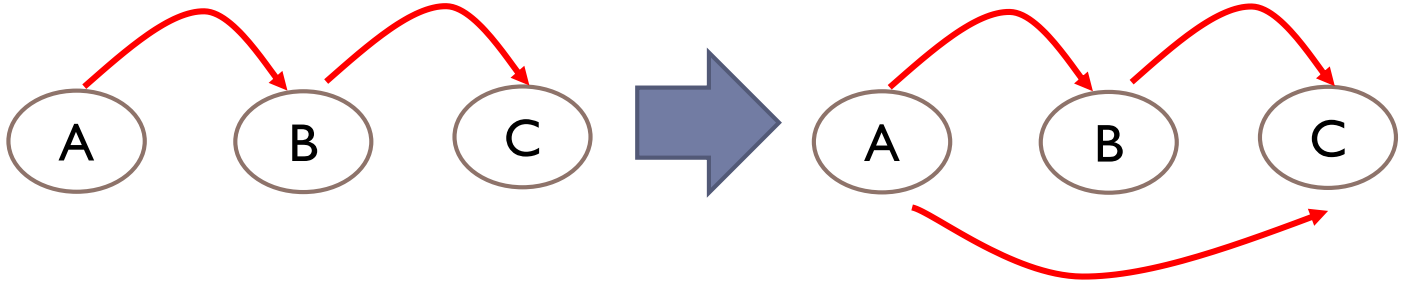
- Reflexivity Rule : ถ้า B เป็นซับเซตของ A ดังนั้น A → B



- Augmentation Rule: ถ้า A → B ดังนั้น AC → BC

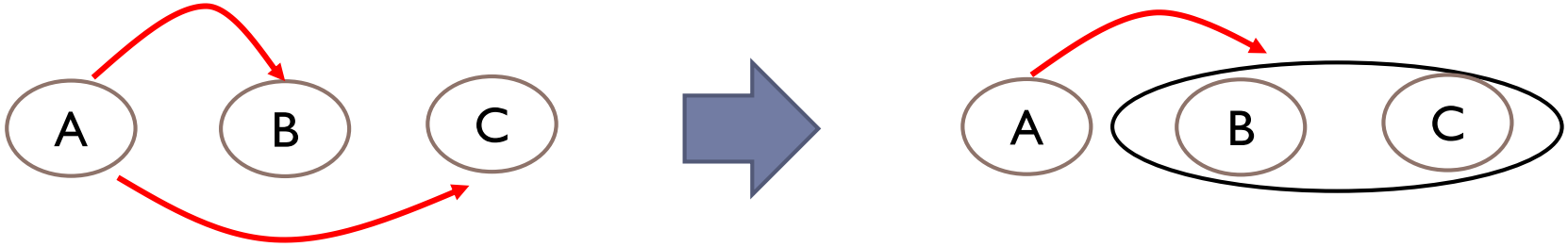


- Transitivity Rule: ถ้า A → B และ B → C ดังนั้น A → C

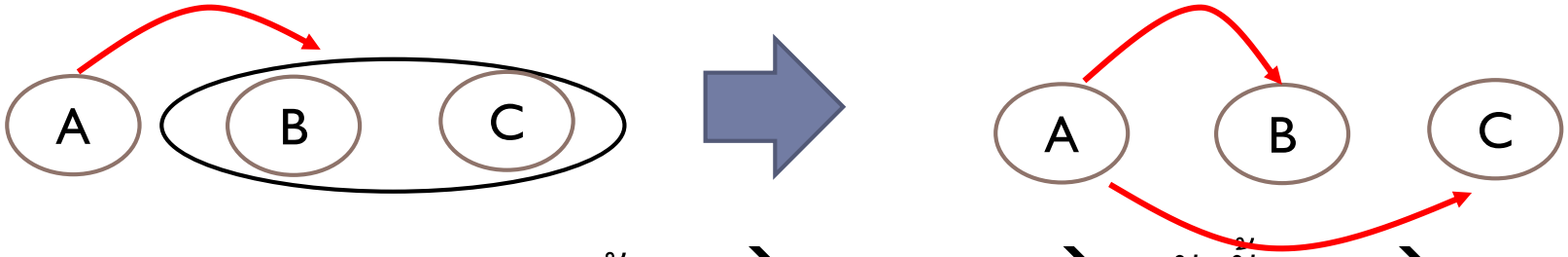


Derived Rules (from Armstrong's Axioms)

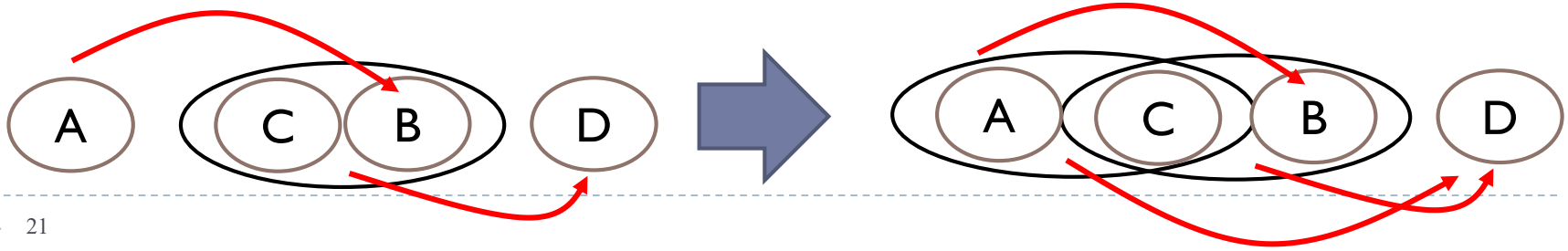
- Union Rule: ถ้า A → B และ A →C ดังนั้น A → B,C



- Decomposition Rule: ถ้า A → B,C ดังนั้น A → B และ A → C

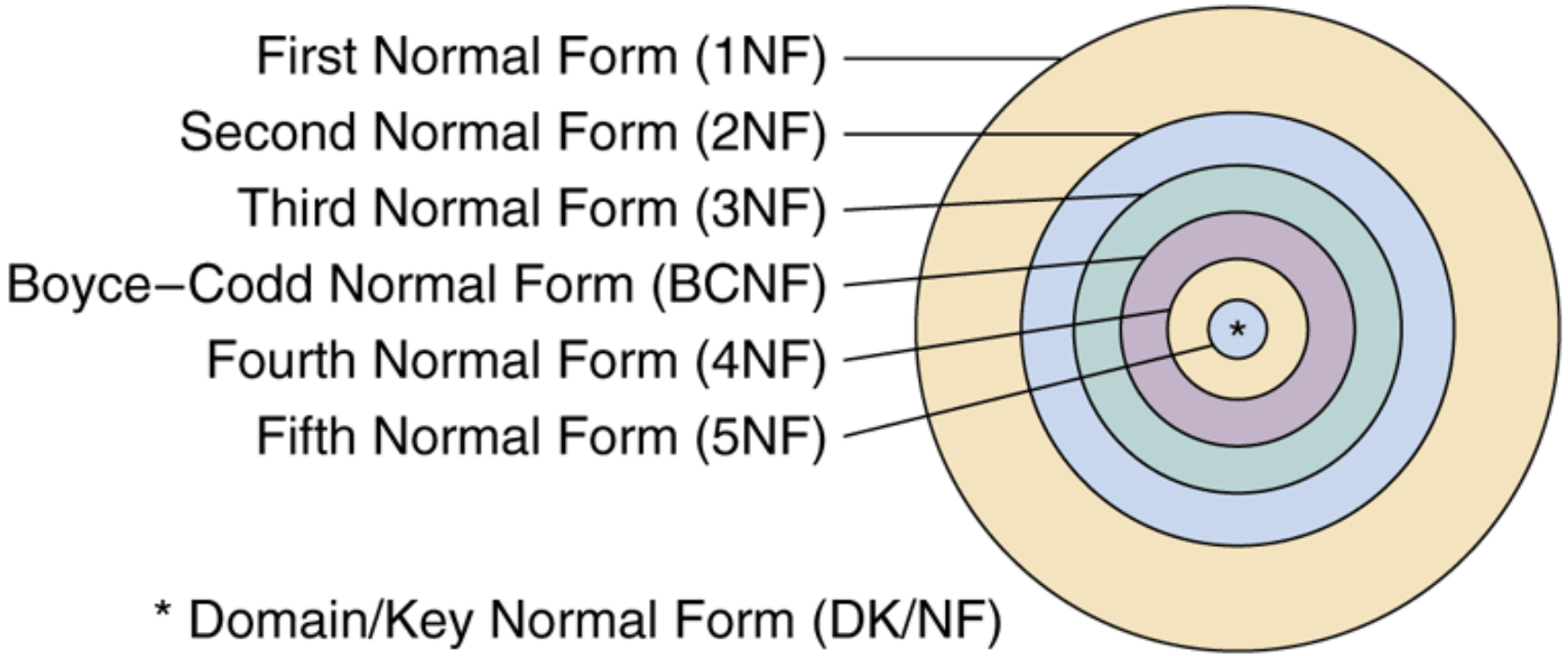


- Psuedotransitivity Rule: ถ้า A →B และ CB →D ดังนั้น AC→D

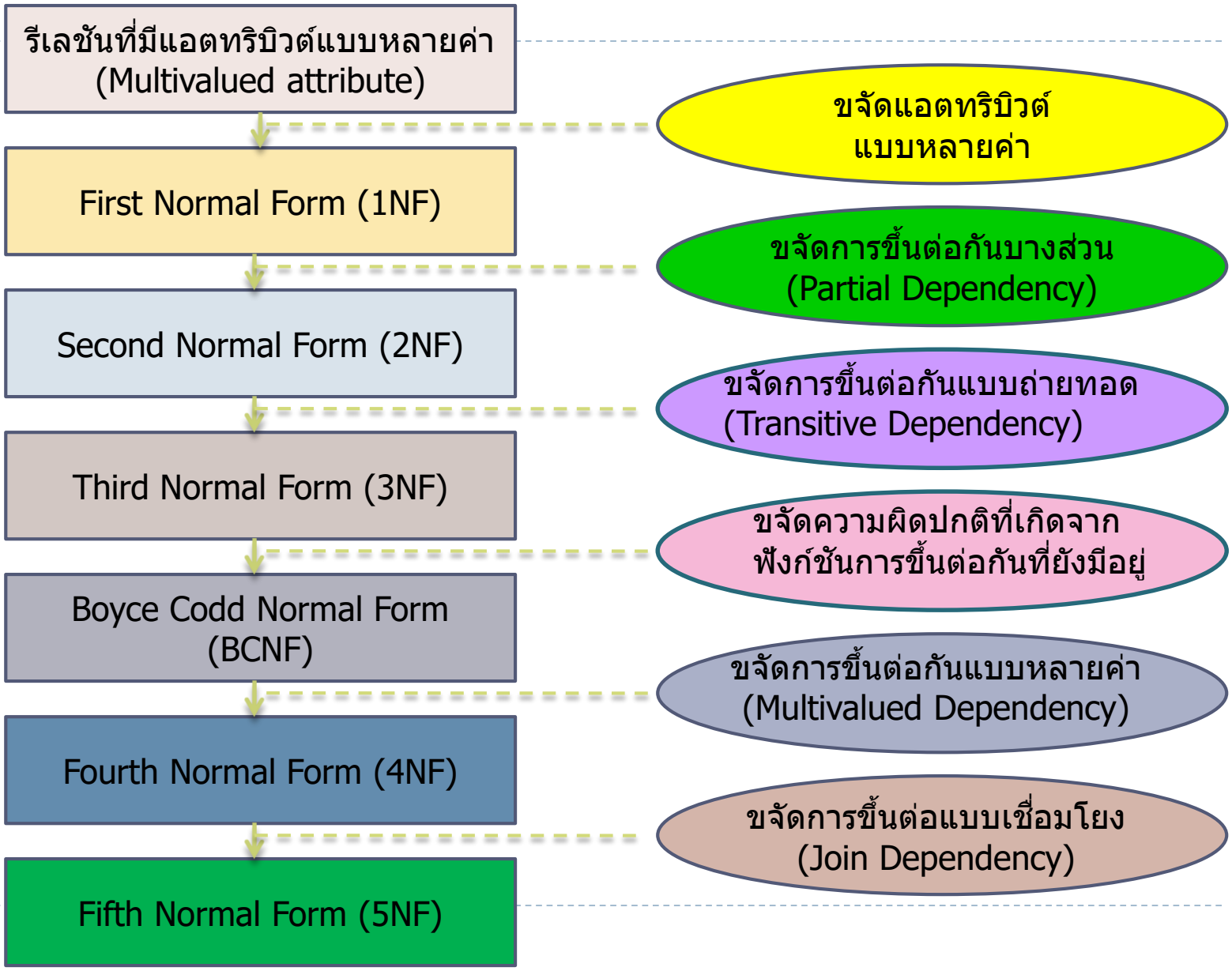


Normalization

Normalization :การปรับบรรทัดฐาน



Step in Normalization



First Normal Form (1NF)

- รีเลชันที่อยู่ในรูปแบบบรรทัดฐานขั้นที่ 1 ก็ต่อเมื่อ รีเลชันนั้นไม่มีกลุ่มข้อมูลซ้ำ(Repeating Group)
- ทุกคอลัมน์เป็น Atomic (1 cell 1 value)

Registration (Relation R)						
Course	Unit	Term	Instructor	Student	Advisor	Office
642	3	2556-1	John	Bell	Miller	521
		2556-1	John	Kent	Ullman	521
		2556-2	David	Brian	White	523
644	3	2556-1	David	Howard	Ullman	521
		2556-2	John	Bell	Miller	521
		2556-2	John	Jordan	Smith	523
700	1	2556-1	Smith	Walker	West	522
		2556-2	Smith	Jordan	Smith	523
		2556-2	Smith	Howard	Ullman	521
799	2	2556-1	West	Walker	West	522
		2556-2	White	Bell	Miller	521

Multivalued Attribute
ปรากฏอยู่ใน Relation

Registration (Relation R)						
Course	Unit	Term	Instructor	Student	Advisor	Office
642	3	2556-1	John	Bell	Miller	521
642	3	2556-1	John	Kent	Ullman	521
642	3	2556-2	David	Brian	White	523
644	3	2556-1	David	Howard	Ullman	521
644	3	2556-2	John	Bell	Miller	521
644	3	2556-2	John	Jordan	Smith	523
700	1	2556-1	Smith	Walker	West	522
700	1	2556-2	Smith	Jordan	Smith	523
700	1	2556-2	Smith	Howard	Ullman	521
799	2	2556-2	West	Walker	West	522
799	2	2556-2	White	Bell	Miller	521

กำจัด

Multivalued Attribute

1 NF (Example)

Customer	Cust. ID	Transactions
Jones	1	Tr. ID: 12890, Date: 14-Oct-2003, Amount: −87 Tr. ID: 12904, Date: 15-Oct-2003, Amount: −50
Wilkins	2	Tr. ID: 12898, Date: 14-Oct-2003, Amount: −21
Stevens	3	Tr. ID: 12907, Date: 15-Oct-2003, Amount: −18 Tr. ID: 14920, Date: 20-Nov-2003, Amount: −70 Tr. ID: 15003, Date: 27-Nov-2003, Amount: −60

1 NF (Example)

Customer	Cust. ID	Tr.ID	Date	Amount
Jones	1	12890	14-Oct-2003	-87
Jones	1	12904	15-Oct-2003	-50
Wilkins	2	12898	14-Oct-2003	-21
Stevens	3	12907	15-Oct-2003	-18
Stevens	3	14920	20-Nov-2003	-70
Stevens	3	15003	27-Nov-2003	-60
Jones	4	16901	27-Nov-2003	-18

Second Normal Form (2NF)

รีเลชันที่อยู่ในรูปแบบบรรทัดฐานขั้นที่ 2 ก็ต่อเมื่อ

- รีเลชันนั้นต้องอยู่ในรูปแบบ 1NF
- ไม่มีแอททริบิวต์ที่มีการขึ้นต่อบางส่วนของคีย์หลัก (Partial Dependency)

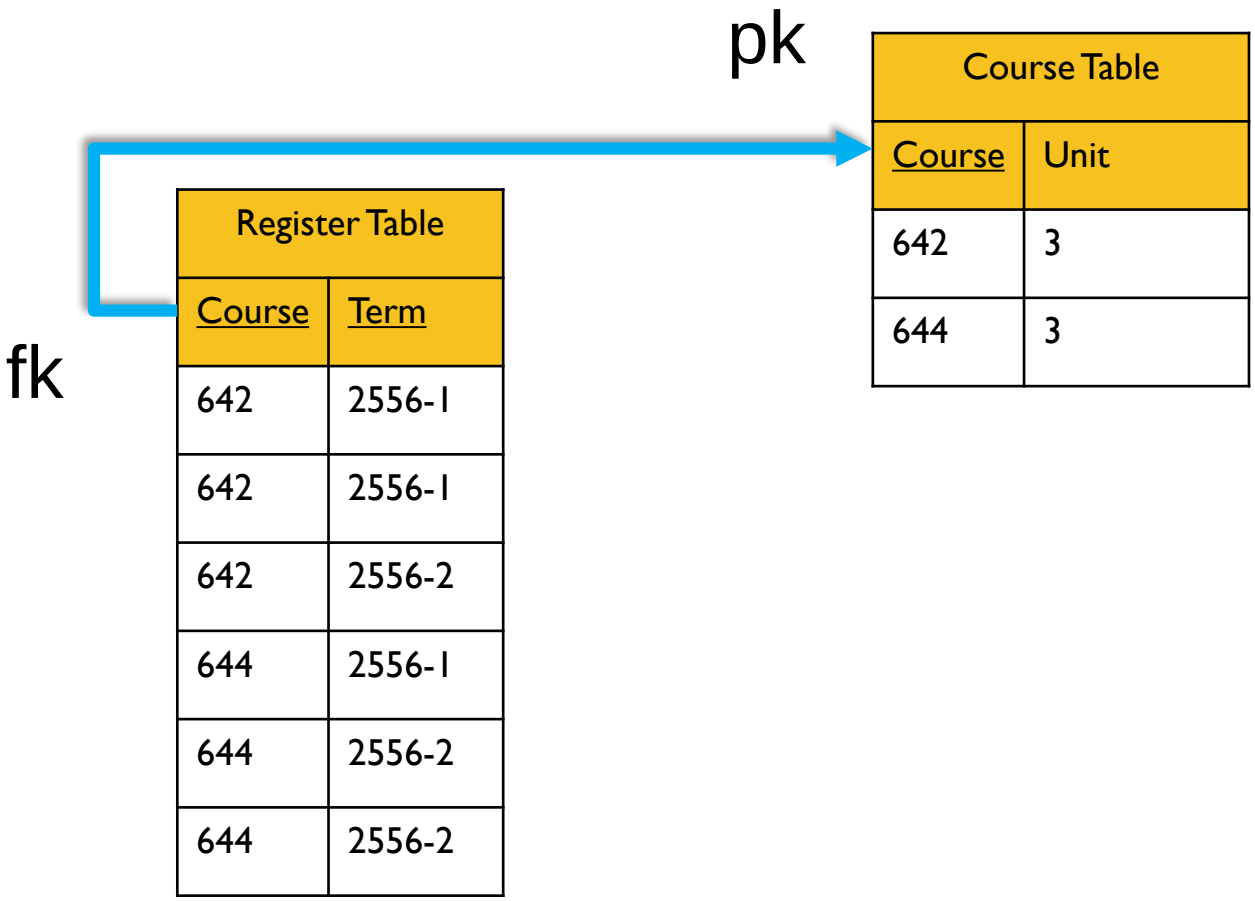
2NF Example

- ตาราง Register มีค่า Unit ที่ซ้ำกัน
- ซึ่ง Unit ถูกกำหนดโดย Course (Course → Unit)

Register Table		
Course	Unit	Term
642	3	2556-1
642	3	2556-1
642	3	2556-2
644	3	2556-1
644	3	2556-2
644	3	2556-2

2NF

- แยกตารางออกมา



2NF (Example)

Electric Toothbrush Models

Manufacturer	Model	Model Full Name	Manufacturer Country
Forte	X-Prime	Forte X-Prime	Italy
Forte	Ultraclean	Forte Ultraclean	Italy
Dent-o-Fresh	EZbrush	Dent-o-Fresh EZbrush	USA
Kobayashi	ST-60	Kobayashi ST-60	Japan
Hoch	Toothmaster	Hoch Toothmaster	Germany
Hoch	X-Prime	Hoch X-Prime	Germany

2NF (Example)

Electric Toothbrush Manufacturers

Manufacturer	Manufacturer Country
Forte	Italy
Dent-o-Fresh	USA
Kobayashi	Japan
Hoch	Germany

Electric Toothbrush Models

Manufacturer	Model	Model Full Name
Forte	X-Prime	Forte X-Prime
Forte	Ultraclean	Forte Ultraclean
Dent-o-Fresh	EZbrush	Dent-o-Fresh EZbrush
Kobayashi	ST-60	Kobayashi ST-60
Hoch	Toothmaster	Hoch Toothmaster
Hoch	X-Prime	Hoch X-Prime

Third Normal Form (3NF)

รีเลชันที่อยู่ในรูปแบบบรรทัดฐานขั้นที่ 3 ก็ต่อเมื่อ

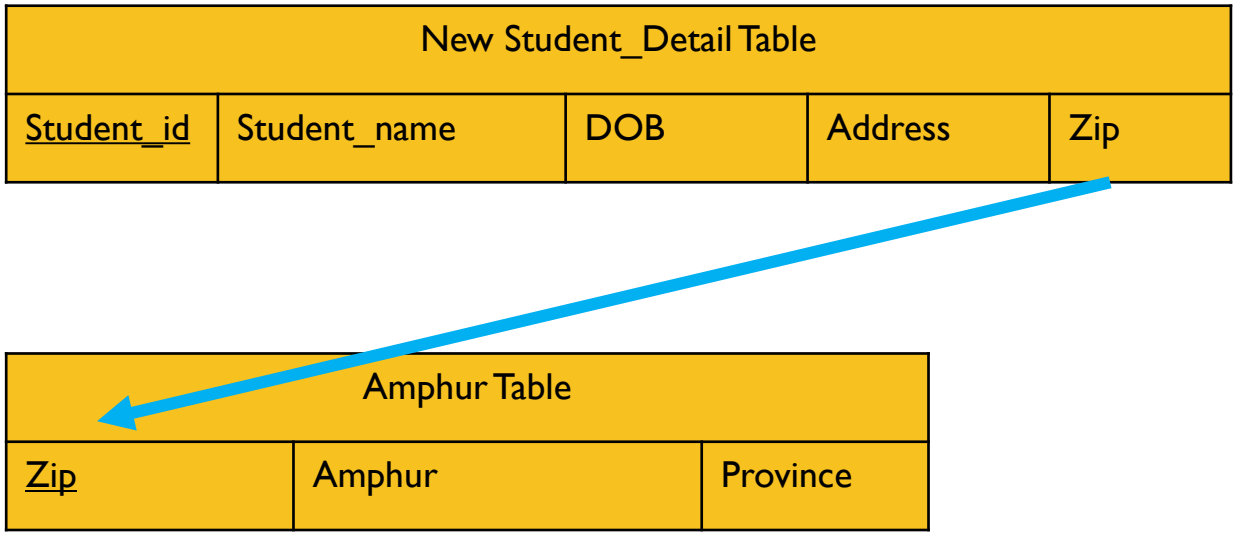
- รีเลชันนั้นต้องอยู่ในรูปแบบ 2NF และ
- ไม่มีแอททริบิวต์ที่ไม่ใช่คีย์หลักสามารถระบุค่าแอททริบิวต์ที่ไม่ใช่คีย์หลักอื่นได้ (Transitive Dependency)

3NF-Example1

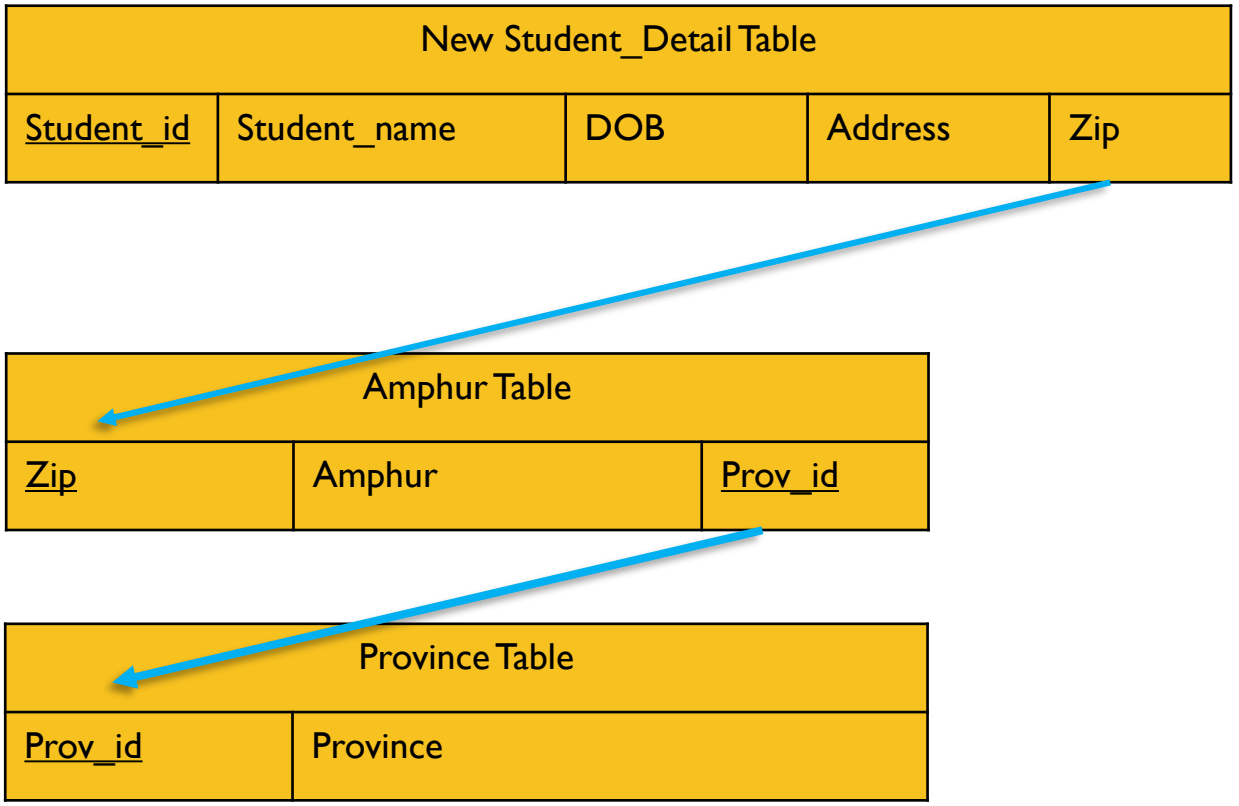
- PK:Student_id
- แต่ Zip → Province, Amphur ซึ่งเป็น Transitive DP

Student_Detail Table						
Student_id	Student_name	DOB	Address	Amphur	Province	Zip

3NF-Example1



3NF-Example1



3NF-Example2

Tournament Winners

Tournament	Year	Winner	Winner Date of Birth
Indiana Invitational	1998	Al Fredrickson	21 July 1975
Cleveland Open	1999	Bob Albertson	28 September 1968
Des Moines Masters	1999	Al Fredrickson	21 July 1975
Indiana Invitational	1999	Chip Masterson	14 March 1977

Tournament + Year → Winner, DOB

Winner → DOB

3NF-Example2

fk pk

Tournament Winners

Tournament	Year	Winner
Indiana Invitational	1998	Al Fredrickson
Cleveland Open	1999	Bob Albertson
Des Moines Masters	1999	Al Fredrickson
Indiana Invitational	1999	Chip Masterson

Winner Dates of Birth

Winner	Date of Birth
Chip Masterson	14 March 1977
Al Fredrickson	21 July 1975
Bob Albertson	28 September 1968

Tournament + Year → Winner

Winner → DOB

3NF-Example3

Customer	Cust. ID	Tr.ID	Date	Amount
Jones	1	12890	14-Oct-2003	-87
Jones	1	12904	15-Oct-2003	-50
Wilkins	2	12898	14-Oct-2003	-21
Stevens	3	12907	15-Oct-2003	-18
Stevens	3	14920	20-Nov-2003	-70
Stevens	3	15003	27-Nov-2003	-60
Jones	4	16908	27-Nov-2003	-18

Tr.ID → Cust.ID, Customer, Date, Amount

Cust.ID → Customer

3NF-Example3

pk

Customer	Cust. ID
Jones	1
Wilkins	2
Stevens	3
Jones	4

Cust.ID → Customer

fk

Cust. ID	Tr.ID	Date	Amount
1	12890	14-Oct-2003	-87
1	12904	15-Oct-2003	-50
2	12898	14-Oct-2003	-21
3	12907	15-Oct-2003	-18
3	14920	20-Nov-2003	-70
3	15003	27-Nov-2003	-60
4	16908	27-Nov-2003	-18

Tr.ID → Cust.ID, Date, Amount

3NF คำนิยามอื่น ๆ

- ทุก FD (ตัวอย่าง A→B) ในทุกรีเลชั่นที่เป็น 3NF ได้ต้อง
  - เป็น trivial FD
  - A เป็น key ของรีเลชั่น
  - หรือทุกค่าใน B เป็นส่วนหนึ่งของ candidate key สำหรับรีเลชั่นนั้น ๆ
- เราสามารถหา 3NF ได้จากการ
  - Lossless-join Decomposition และ
  - Dependency Preservation Decomposition

Boyce Codd Normal Form (BCNF)

- รูปแบบของ 3NF ขั้นสูง
- ทุก FD (ตัวอย่าง A→B) ในทุกรีเลชั่นที่เป็น BCNF ได้ต้อง
  - เป็น trivial FD หรือ
  - A เป็น key ของรีเลชั่น
  - ~~*ทุกค่าใน B เป็นส่วนหนึ่งของ candidate key สำหรับรีเลชั่นนั้น ๆ*~~
- เราสามารถหา BCNF ได้จากการ
  - Lossless-join Decomposition และ
  - Dependency Preservation Decomposition

3NF vs BCNF

- รีเลชั่นใดถ้ามีลักษณะ BCNF แล้วจะเป็น 3NF แน่นอน
- บางครั้งการทำ 3NF จะกลายเป็น BCNF ไปโดยปริยาย
- ดังตัวอย่าง ต่อไปนี้

BCNF-Example

pk

Customer	Cust. ID
Jones	1
Wilkins	2
Stevens	3
Jones	4

Cust.ID → Customer

fk

Cust. ID	Tr.ID	Date	Amount
1	12890	14-Oct-2003	-87
1	12904	15-Oct-2003	-50
2	12898	14-Oct-2003	-21
3	12907	15-Oct-2003	-18
3	14920	20-Nov-2003	-70
3	15003	27-Nov-2003	-60
4	16908	27-Nov-2003	-18

Tr.ID → Cust.ID, Date, Amount

ทำ 3NF แล้วกลายเป็น BCNF โดยอัตโนมัติ

BCNF-Example

fk pk

Tournament Winners

Tournament	Year	Winner
Indiana Invitational	1998	Al Fredrickson
Cleveland Open	1999	Bob Albertson
Des Moines Masters	1999	Al Fredrickson
Indiana Invitational	1999	Chip Masterson

Winner Dates of Birth

Winner	Date of Birth
Chip Masterson	14 March 1977
Al Fredrickson	21 July 1975
Bob Albertson	28 September 1968

Tournament + Year → Winner

Winner → DOB

ทำ 3NF แล้วกลายเป็น BCNF โดยอัตโนมัติ

Boyce/Codd Normal Form : BCNF

รูปแบบของรีเลชันที่ต้องผ่านการปรับให้เป็น BCNF

- บางส่วนของ key ไปขึ้นกับ key อื่น
- รีเลชันที่มีคีย์คู่แข่งหลายคีย์
- คีย์คู่แข่งเป็นคีย์ผสม
- คีย์คู่แข่งมีบางส่วนซ้ำซ้อนกัน

3NF not meet BCNF

- บางครั้งการทำ 3NF จะกลายเป็น BCNF ไปโดยปริยาย
- แต่บางครั้งอาจไม่เป็น BCNF

3NF not meet BCNF

Client , Office → Account

Account → Office

บางส่วนของ key ขึ้นกับ key อื่น

for every FD $A \rightarrow B$

- เป็น trivial FD
- *A* เป็น key ของรีเลชั่น
- ทุกค่าใน B เป็นส่วนหนึ่งของ candidate key สำหรับรีเลชั่นนั้น ๆ

Account	Client	Office
A	Joe	1
B	Mary	1
A	John	1
C	Joe	2

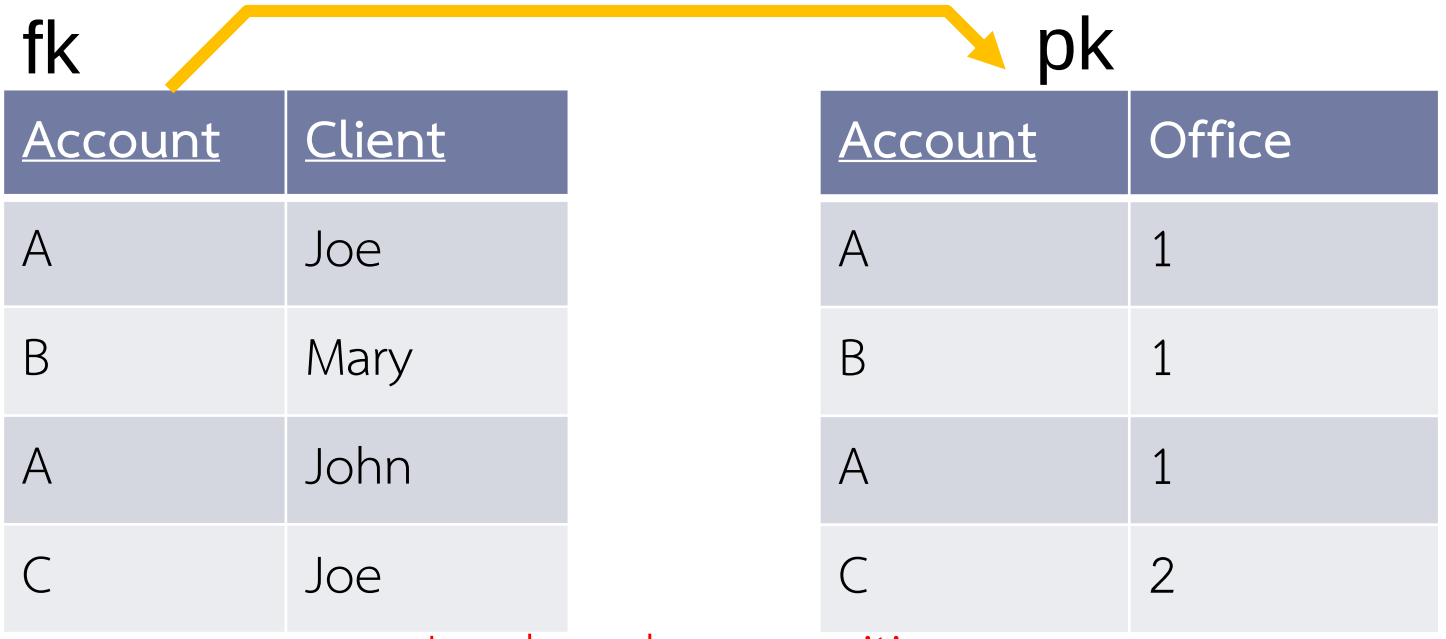
3NF not meet BCNF

for every FD $A \rightarrow B$

- เป็น trivial FD
- A เป็น key ของรีเลชั่น

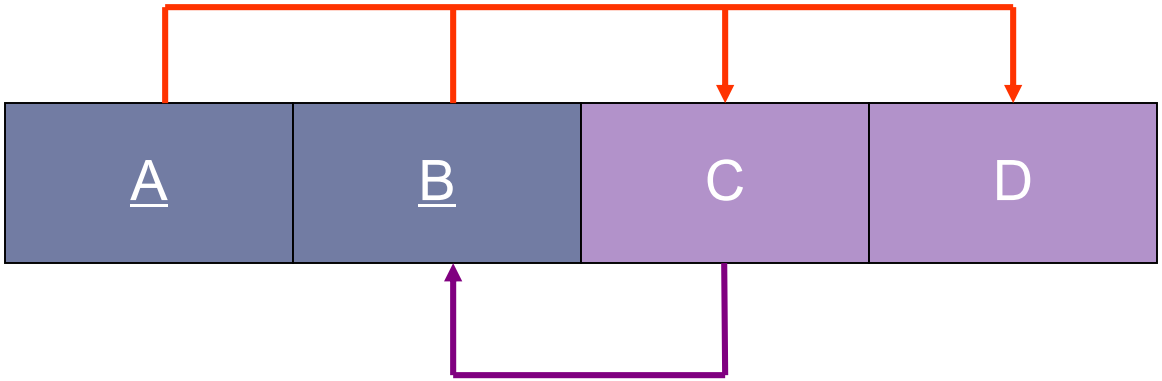
- 3NF has some redundancy

  BCNF does not
- but BCNF is not DPD (ดูเนื้อหา DPD)



Lossless decomposition

3NF not meet BCNF(2)

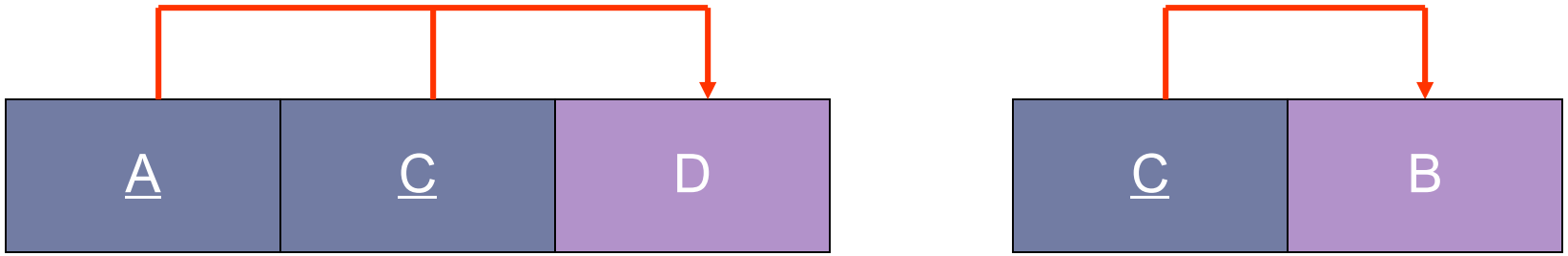


A + B → C , D

C → B

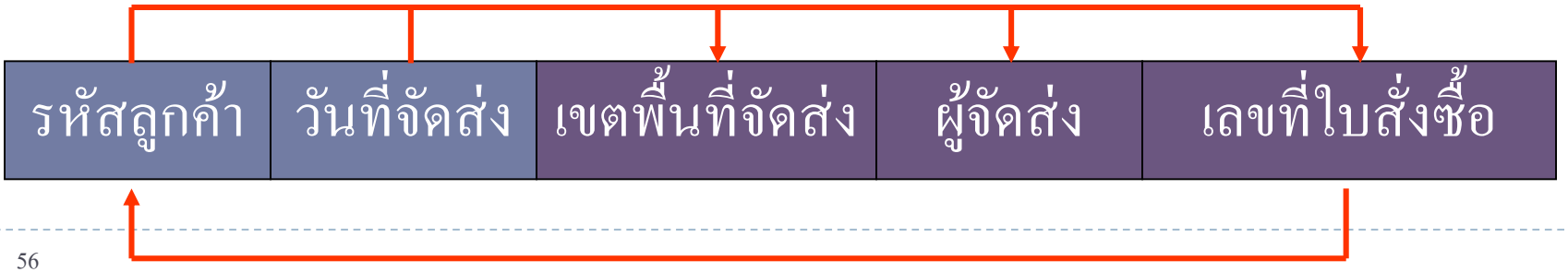
3NF not meet BCNF(2)

- รีเลชันเดิมเปลี่ยนแอททริบิวต์ของคีย์หลักจาก B เป็น C (เพราะเมื่อ B ขึ้นกับ C แสดงว่าถ้าทราบค่า C ก็จะสามารถระบุค่า B ได้เช่นเดียวกัน) และตัดแอททริบิวต์ B ออกจากรีเลชัน
- สร้างรีเลชันใหม่โดยมี C เป็นคีย์หลัก และ B เป็นแอททริบิวต์ที่ขึ้นกับค่า C

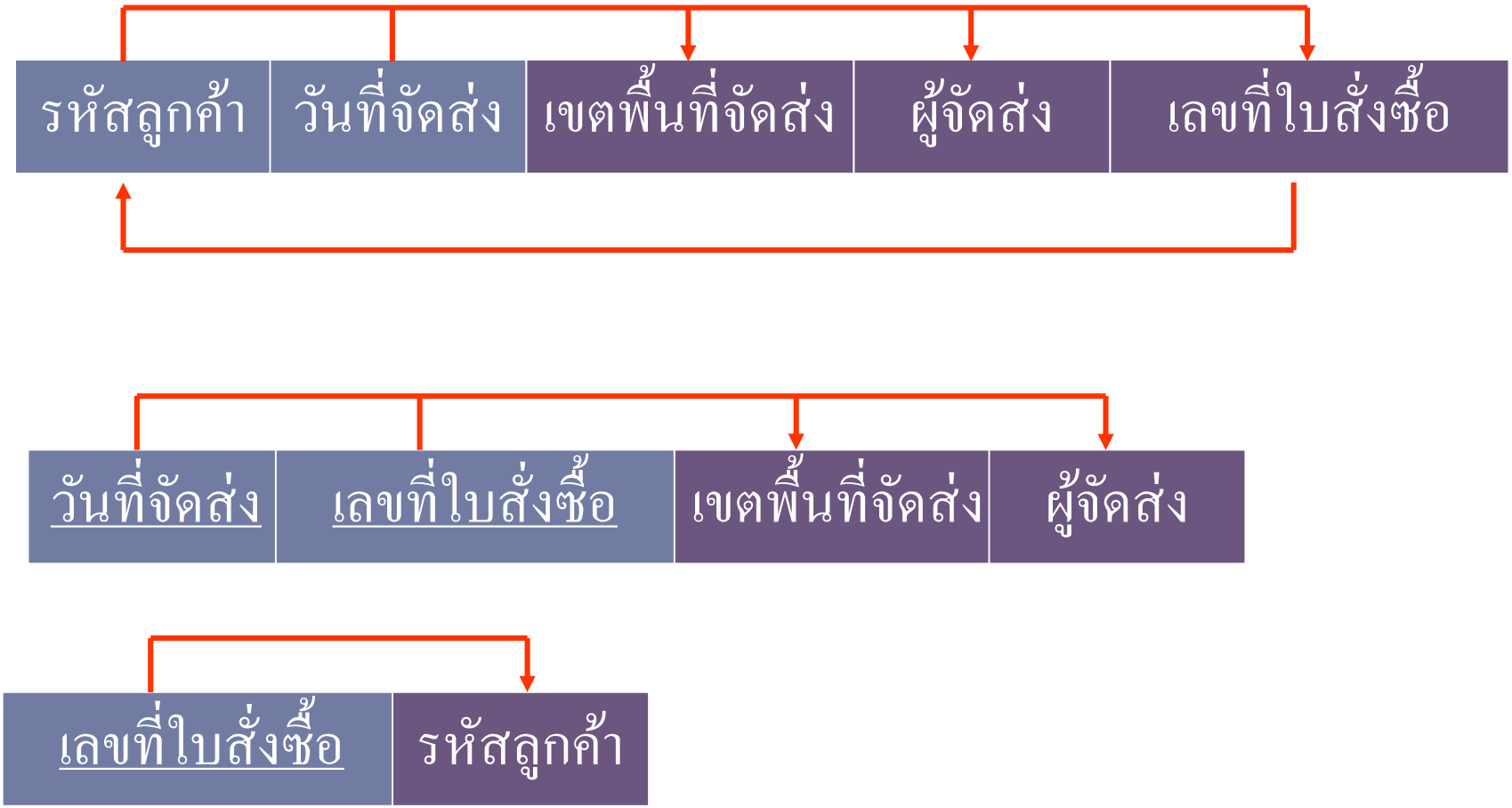


การจัดส่งสินค้า

รหัสลูกค้า	วันที่จัดส่ง	เขตพื้นที่จัดส่ง	ผู้จัดส่ง	เลขที่ใบสั่งซื้อ
111	25/6/2550	A1	สมชาย	B01
222	28/6/2550	A2	ยิ่งยง	B02
111	28/6/2550	A1	ยิ่งยง	B01
333	2/7/2550	A1	สมชาย	B03
222	2/7/2550	A2	ชาติชาย	B04
444	10/7/2550	A2	สมชาย	B05



3NF not meet BCNF(2)



รีเลชันที่อยู่ในรูป BCNF

เลขที่ใบสั่งซื้อ	วันที่จัดส่ง	เขตพื้นที่จัดส่ง	ผู้จัดส่ง
B01	25/6/2550	A1	สมชาย
B02	28/6/2550	A2	ยิ่งยง
B01	28/6/2550	A1	ยิ่งยง
B03	2/7/2550	A1	สมชาย
B04	2/7/2550	A2	ชาติชาย
B05	10/7/2550	A2	สมชาย

เลขที่ใบสั่งซื้อ	รหัสลูกค้า
B01	111
B02	222
B03	333
B04	222
B05	444

3NF but not meet BCNF(3)

Patient No	Patient Name	Appointment Id	Time	Doctor
1	John	0	09:00	Zorro
2	Kerr	0	09:00	Kane
3	Adam	1	10:00	Zorro
4	Robert	0	13:00	Kane
5	Zane	1	14:00	Zorro

PatNo → PatName

PatNo, appID → Time, Doctor

Time → appID

PK : PatNo + appID or PatNo + Time

R1(<u>PatNo</u>, <u>Time</u>, Doctor)

R2(<u>PatNo</u>, PatName)

R3(<u>Time</u>, AppID)

บางส่วนของ key ขึ้นกับ key อื่น

Fourth Normal Form : 4NF

รีเลชันที่อยู่ในรูปแบบบรรทัดฐานขั้นที่ 4(4NF) ก็ต่อเมื่อ

- รีเลชันนั้นต้องอยู่ในรูป (BCNF)
- ไม่มีการขึ้นต่อกันแบบหลายค่า (Multivalued Dependency)

Fifth Normal Form : 5NF

รีเลชันที่อยู่ในรูปแบบบรรทัดฐานขั้นที่ 5(5NF) ก็ต่อเมื่อ

- รีเลชันนั้นต้องอยู่ในรูป 4(4NF)
- เมื่อแบ่งรีเลชันออกเป็นรีเลชันย่อยแล้วสามารถ join กลับได้ค่าเท่ากับรีเลชันเดิม ไม่มีรายการข้อมูลเกิน (spurious tuple) จะกล่าวว่ารีเลชันย่อยๆนั้นอยู่ในรูป 5NF

Lossless-join Decomposition

Lossless-join Decomposition (LJD)

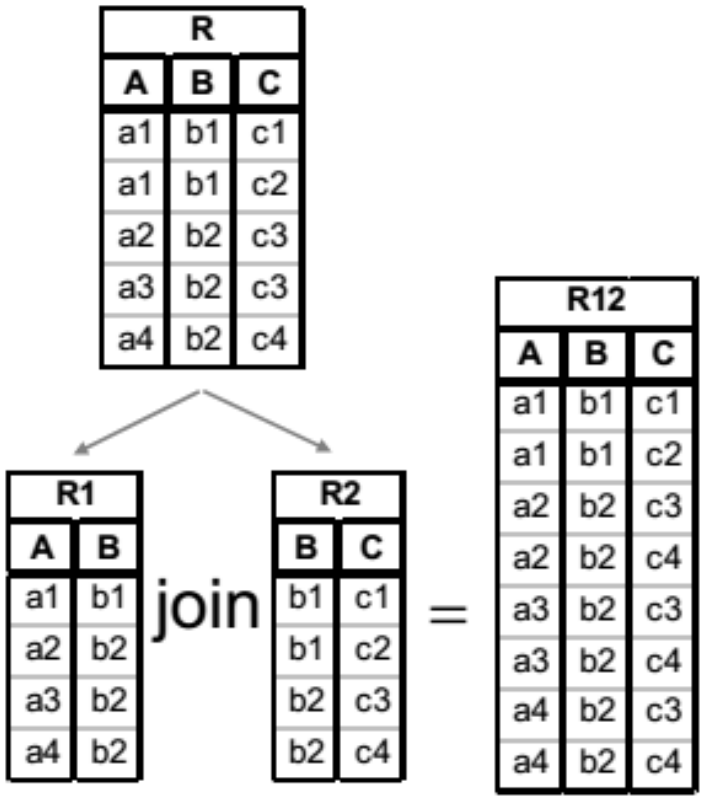
- หากทำการแยกตารางออกจากกันแล้ว (decomposition) เมื่อรวมกัน (join) กลับคืนมาจะสามารถทำให้ข้อมูลกลับมาสู่โครงสร้างเดิมได้
  - $R1 \cap R2 \rightarrow R1$
  - $R1 \cap R2 \rightarrow R2$
- แอททริบิวต์ต้องอยู่ครบดังเดิม

ตัวอย่าง LJD

R(CUTISAO); S→A→O; C→U; CT→I

- LJDs
  - R1(CUTISA), R2(SO)
    - S → R2
    - แต่ยังเป็น decomposition ที่ไม่ดี
  - R1(CU), R2(CTISAO)
    - C →R1
    - decomposition ที่ดี
  - R1(CU), R2(CTI), R3(CTS), R4(SA), R5(AO)
    - เมื่อทำ LJD หลาย ๆ ครั้ง
- ไม่ใช่ LJD
  - R1(CUTI) , R2(SAO)
    - R1 ∩ R2 = null
  - R1(CI), R2(UTISAO)
    - I ไม่ →R1 และ I ไม่ → R2

ตัวอย่างการแยกตารางโดยไม่ใช้ LJD



- ตัวอย่าง
  - R(A,B,C)
  - F = {A→B, C→B}
- แยกโดยไม่ใช้ LJD
  - R1(A,B); F1 = {A→B}
  - R2(B,C); F2 = {C→B}
- ผลที่ได้
  - R1 join R2 <> R

ตัวอย่างการแยกตารางโดยไม่ใช้ LJD

- กำหนด R(A, B, C, D, E, F)
- โดยแยกตารางได้ดังนี้ R1(A,B,C) R2(C, D, E, F)
- R1 ∩ R2 = {C}
- {C} → {A, B, C} ไม่ได้
- {C} → {C, D, E, F} ไม่ได้
- ไม่เป็น LJD

Dependency Preservation Decomposition

Dependency Preservation Decomposition (DPD)

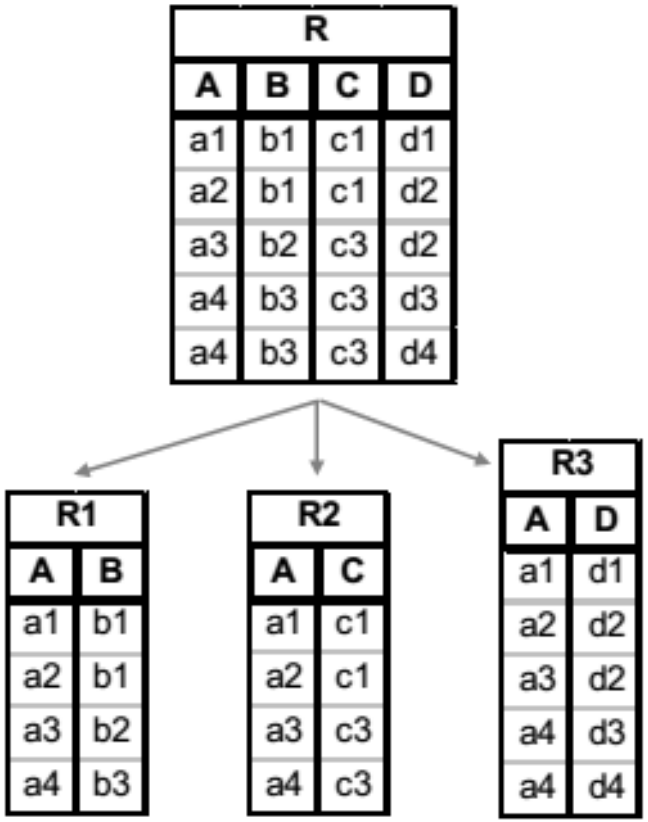
- ทุกเงื่อนไขหรือข้อกำหนดเดิม (Constraint) จะต้องอยู่ครบถ้วนหลังจากการแยกตารางออกจากกัน (Decomposition)

Example of DPD

R(CUTISAO); S→A→O; C→U; CT→I

- Preserve Dependencies
  - R1(CUTI), R2(SAO)
    - ทุก dependencies อยู่ครบถ้วน แต่ไม่รักษากฎ LJD
  - R1(CU), R2(CTI), R3(CTS), R4(SA), R5(AO)
    - ทุก dependencies อยู่ครบถ้วน
- Not Preserve Dependencies
  - R1(CUTISA), R2(SO)
    - A → O หายไป
- PD & LJD
  - R1(CUI), R2(CTISAO)
  - ยังเป็นการแยกตารางที่ดีไม่ได้ ทำไม ?

ตัวอย่างการแยกตัวที่ไม่รักษากฎ DPD



- ตัวอย่าง
  - R(A,B,C,D)
  - F={A→B, B→C}
- ไม่ใช่ DP
  - R1(A,B); F1 = {A→B}
  - R2(A,C); F2 = {A→C}
  - R3(A,D)
- ผลกระทบที่ไม่ต้องการ
  - ไม่สามารถบังคับใช้ FD: B→C โดยไม่ต้อง join R1 และ R2

ตัวอย่างการแยกตัวที่ไม่รักษากฎ DPD

- กำหนด R(C, U, T, I, S, A, O);
- FD ได้แก่ S→A; A→O; C→U; CT→I

R1(S,O)

A→O ??

R2(C, U, T, I, S, A)
C→U
CT→I
S→A

- เป็น LJD แต่ ไม่เป็น DPD..

End of Relational Database Design