

13 장. 관계 데이터베이스의 함수적 종속성과 정규화 기본 이론

Contents

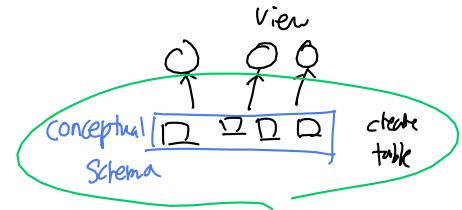
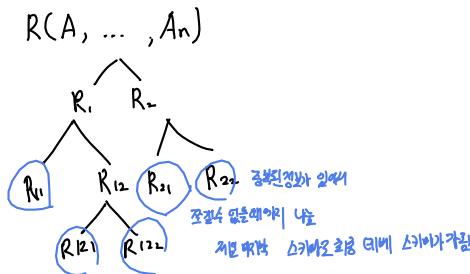
- 릴레이션 스키마를 설계하는 몇 가지 개략적인 지침
- **함수적 종속성 (functional dependencies, FDs)**
- 기본 키를 기반으로 한 정규형
- 제 2 정규형과 제 3 정규형의 일반적인 정의
- BCNF (Boyce-Codd Normal Form)

Relational Database design의 가장 중요한 목표 2개

- Information preservation
- minimum redundancy

릴레이션 스키마를 설계하는 몇 가지 개략적인 지침

- 관계형 데이터베이스 설계란?
 - “좋은” 릴레이션 스키마를 생성하기 위하여 **애트리뷰트들을 그루핑하는 과정**
 - “좋은” 릴레이션에 대한 기준은?
- 릴레이션 스키마의 두 가지 수준
 - 논리적인 “사용자 뷰(user view)” 수준
 - 저장이 되는 “기본 릴레이션(base relation)” 수준
- 데이터베이스 설계는 주로 기본 릴레이션을 대상으로 함
- 내용
 - 여기서는 좋은 릴레이션 설계에 관한 개괄적인 지침을 논한 후, 함수적 종속성과 정규형 개념에 관해 논의함
 - 1NF (제 1 정규형)
 - 2NF (제 2 정규형)
 - 3NF (제 3 정규형)
 - BCNF (Boyce-Codd 정규형)



one information at one place

No Redundancy

① EMP(ssn, name, ..., salary)

② EMP(ssn, name, name) *

EMP_SAL(ssn, salary) 1..1

③ Space를 약제 사용한다.

④ 계약 조건에 모든 충족 가능해야 한다.

릴레이션 애트리뷰트들의 의미

설계자는 데이터베이스에 포함될 전체 애트리뷰트들을 대상으로 실세계에서 어떤 연관성이 있는 애트리뷰트들을 묶어서 하나의 릴레이션 스키마를 만들어 나간다.

- 여러 엔티티(EMPLOYEE, DEPARTMENT, PROJECT)의 애트리뷰트들이 하나의 릴레이션에 혼합되면 안된다.
 $\text{foreign keys 만에 서로 다른 entity들을}$
 참조할 수 있어야 한다
- 다른 엔티티를 참조하기 위해서는 외래키만을 사용해야 한다.

예 :

- 잘 설계된 경우 (Fig 15.1)
- 데이터베이스 인스턴스 (Fig 15.2)
- 여러 엔티티의 속성들이 하나의 릴레이션에 혼합되어 문제 발생 (Fig 15.3)

Figure 15.1

A simplified COMPANY relational database schema.

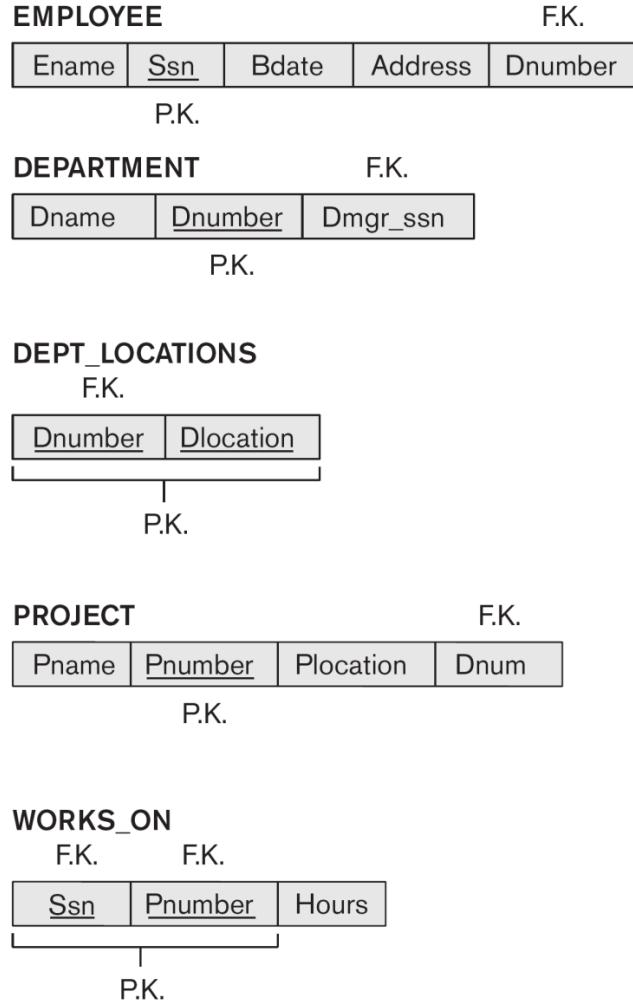


Figure 15.2

Sample database state for the relational database schema in Figure 15.1.

EMPLOYEE

| Ename | Ssn | Bdate | Address | Dnumber |
|----------------------|-----------|------------|--------------------------|---------|
| Smith, John B. | 123456789 | 1965-01-09 | 731 Fondren, Houston, TX | 5 |
| Wong, Franklin T. | 333445555 | 1955-12-08 | 638 Voss, Houston, TX | 5 |
| Zelaya, Alicia J. | 999887777 | 1968-07-19 | 3321 Castle, Spring, TX | 4 |
| Wallace, Jennifer S. | 987654321 | 1941-06-20 | 291Berry, Bellaire, TX | 4 |
| Narayan, Ramesh K. | 666884444 | 1962-09-15 | 975 Fire Oak, Humble, TX | 5 |
| English, Joyce A. | 453453453 | 1972-07-31 | 5631 Rice, Houston, TX | 5 |
| Jabbar, Ahmad V. | 987987987 | 1969-03-29 | 980 Dallas, Houston, TX | 4 |
| Borg, James E. | 888665555 | 1937-11-10 | 450 Stone, Houston, TX | 1 |

DEPARTMENT

| Dname | Dnumber | Dmgr_ssn |
|----------------|---------|-----------|
| Research | 5 | 333445555 |
| Administration | 4 | 987654321 |
| Headquarters | 1 | 888665555 |

DEPT_LOCATIONS

| Dnumber | Dlocation |
|---------|-----------|
| 1 | Houston |
| 4 | Stafford |
| 5 | Bellaire |
| 5 | Sugarland |
| 5 | Houston |

WORKS_ON

| Ssn | Pnumber | Hours |
|-----------|---------|-------|
| 123456789 | 1 | 32.5 |
| 123456789 | 2 | 7.5 |
| 666884444 | 3 | 40.0 |
| 453453453 | 1 | 20.0 |
| 453453453 | 2 | 20.0 |
| 333445555 | 2 | 10.0 |
| 333445555 | 3 | 10.0 |
| 333445555 | 10 | 10.0 |
| 333445555 | 20 | 10.0 |
| 999887777 | 30 | 30.0 |
| 999887777 | 10 | 10.0 |
| 987987987 | 10 | 35.0 |
| 987987987 | 30 | 5.0 |
| 987654321 | 30 | 20.0 |
| 987654321 | 20 | 15.0 |
| 888665555 | 20 | Null |

PROJECT

| Pname | Pnumber | Plocation | Dnum |
|-----------------|---------|-----------|------|
| ProductX | 1 | Bellaire | 5 |
| ProductY | 2 | Sugarland | 5 |
| ProductZ | 3 | Houston | 5 |
| Computerization | 10 | Stafford | 4 |
| Reorganization | 20 | Houston | 1 |
| Newbenefits | 30 | Stafford | 4 |

Figure 15.3

Two relation schemas suffering from update anomalies. (a) EMP_DEPT and (b) EMP_PROJ.

(a)

EMP_DEPT

| Ename | Ssn | Bdate | Address | Dnumber | Dname | Dmgr_ssn |
|-------|-----|-------|---------|---------|-------|----------|
| | | | | | | |

(b)

EMP_PROJ

| Ssn | Pnumber | Hours | Ename | Pname | Plocation |
|-----|---------|-------|-------|-------|-----------|
| | | | | | |
| | | | | | |
| | | | | | |

| | | | | | |
|-----|--|--|--|--|--|
| FD1 | | | | | |
| FD2 | | | | | |
| FD3 | | | | | |

Insert anomaly

1. Employee 가 할당되지 않은 project 정보는 Insert 할 수 없다.
Insert 하기 되면 Ssn의 사이즈로, Entity Constraint 위반

2. Project를 부여받지 않은 employee 정보는 Insert 할 수 있다.
Insert 하기 되면 Pnumber가 Null=1로, Entity constraint 위반

Delete Anomaly \Rightarrow 2개의 다른 정보들의 함께
제거되어 발생하는 문제

- Project가 삭제되면 해당 project에 참여하고 있는 모든 employees 정보 또한 삭제될 것이다.
- 어떤 employee가 어떤 project에 참여하는 유일한 직원인 경우, 해당 employee 정보를 삭제한 그 project 정보 또한 삭제되게 된다.

튜플에서 중복된 정보와 이상(anomaly)

하나의 릴레이션에 하나 이상의 엔티티의 애트리뷰트들을 혼합하는 것은 여러 가지 문제를 일으킨다. (Fig 15.4.)

- 정보가 중복 저장되며, 저장 공간을 낭비하게 된다 (Fig 15.2의 EMPLOYEE와 DEPARTMENT \leftrightarrow 15.3 및 15.4의 EMP_DEPT 비교)
- 갱신 이상이 발생하게 된다; 동일한 정보를 한 릴레이션에는 변경하고, 나머지 릴레이션에서는 변경하지 않은 경우 어느 것이 정확한지 알 수 없게 된다.
- 갱신 이상의 종류
 - 삽입 이상 (insertion anomalies)
 - EMP_DEPT에 객체를 삽입할 때 부서가 정해지지 않은 직원이나 직원이 없는 부서를 insert 하는데 문제가 발생함
 - 삭제 이상 (deletion anomalies)
 - 부서의 마지막 직원을 삭제하면 부서 정보도 없어짐
 - 수정 이상 (modification anomalies)
 - 부서 정보를 변경하면 부서의 모든 직원 투플에서 동일하게 변경해야 함

project number 10의 name은

"computerization" $\&$ "Customer-Accounting"으로 변경하는 경우

project number 10에 종사하는 모든 employees를 update 해야 한다.

일관성의 원칙

abnormal 비정상적

anomaly 비정상

one information one place 原則

5번처럼 이름 Research → 4번만 복중임.

Redundancy

| EMP_DEPT | | | | | | |
|----------------------|-----------|------------|--------------------------|---------|----------------|-----------|
| Ename | Ssn | Bdate | Address | Dnumber | Dname | Dmgr_ssn |
| Smith, John B. | 123456789 | 1965-01-09 | 731 Fondren, Houston, TX | 5 | Research | 333445555 |
| Wong, Franklin T. | 333445555 | 1955-12-08 | 638 Voss, Houston, TX | 5 | Research | 333445555 |
| Zelaya, Alicia J. | 999887777 | 1968-07-19 | 3321 Castle, Spring, TX | 4 | Administration | 987654321 |
| Wallace, Jennifer S. | 987654321 | 1941-06-20 | 291 Berry, Bellaire, TX | 4 | Administration | 987654321 |
| Narayan, Ramesh K. | 666884444 | 1962-09-15 | 975 FireOak, Humble, TX | 5 | Research | 333445555 |
| English, Joyce A. | 453453453 | 1972-07-31 | 5631 Rice, Houston, TX | 5 | Research | 333445555 |
| Jabbar, Ahmad V. | 987987987 | 1969-03-29 | 980 Dallas, Houston, TX | 4 | Administration | 987654321 |
| Borg, James E. | 888665555 | 1937-11-10 | 450 Stone, Houston, TX | 1 | Headquarters | 888665555 |

| EMP_PROJ | | | | | | |
|-----------|---------|-------|----------------------|-----------------|-----------|---------|
| Ssn | Pnumber | Hours | Ename | Pname | Plocation | Pnumber |
| 123456789 | 1 | 32.5 | Smith, John B. | ProductX | Bellaire | 1 |
| 123456789 | 2 | 7.5 | Smith, John B. | ProductY | Sugarland | 2 |
| 666884444 | 3 | 40.0 | Narayan, Ramesh K. | ProductZ | Houston | 3 |
| 453453453 | 1 | 20.0 | English, Joyce A. | ProductX | Bellaire | 4 |
| 453453453 | 2 | 20.0 | English, Joyce A. | ProductY | Sugarland | 5 |
| 333445555 | 2 | 10.0 | Wong, Franklin T. | ProductY | Sugarland | 6 |
| 333445555 | 3 | 10.0 | Wong, Franklin T. | ProductZ | Houston | 7 |
| 333445555 | 10 | 10.0 | Wong, Franklin T. | Computerization | Stafford | 8 |
| 333445555 | 20 | 10.0 | Wong, Franklin T. | Reorganization | Houston | 9 |
| 999887777 | 30 | 30.0 | Zelaya, Alicia J. | Newbenefits | Stafford | 10 |
| 999887777 | 10 | 10.0 | Zelaya, Alicia J. | Computerization | Stafford | 11 |
| 987987987 | 10 | 35.0 | Jabbar, Ahmad V. | Computerization | Stafford | 12 |
| 987987987 | 30 | 5.0 | Jabbar, Ahmad V. | Newbenefits | Stafford | 13 |
| 987654321 | 30 | 20.0 | Wallace, Jennifer S. | Newbenefits | Stafford | 14 |
| 987654321 | 20 | 15.0 | Wallace, Jennifer S. | Reorganization | Houston | 15 |
| 888665555 | 20 | Null | Borg, James E. | Reorganization | Houston | 16 |

Figure 15.4

Sample states for EMP_DEPT and EMP_PROJ resulting from applying NATURAL JOIN to the relations in Figure 15.2. These may be stored as base relations for performance reasons.

튜플에서 널 값

Null

- 릴레이션의 튜플들이 널 값을 가지지 않도록 설계해야 함
 - 널 값은 저장 단계에서 공간을 낭비하게 되고 이미 값이 있으면 의미를 명확히 하기 어렵다.
 - 논리적 차원에서는 조인 연산들을 지정하기 힘들고 해석의 어려움
 - 애트리뷰트들의 의미를 이해하기 어려움
 - COUNT나 AVG와 같은 집단 함수들이 적용되었을 때 널 값의 해석이 모호함
 - 널 값은 다음과 같이 여러 가지로 해석이 가능함
 - 그 애트리뷰트가 이 튜플에는 적용되지 않는다. (존재 여부를 모른다)
 - 이 튜플에서 애트리뷰트의 값이 아직 알려져 있지 않다 (존재하지만 모른다).
 - 애트리뷰트 값이 알려져 있지만 DB에 기록되지는 않았다.
 - 모든 널 값을 동일하게 표현하면 널 값이 갖는 여러 의미를 훼손하게 된다.

튜플에서 널 값

- 널 값의 방지 기법 - 릴레이션의 분리

- 널 값이 많이 나타나는 애트리뷰트들은 별개의 릴레이션으로 분리함

| 1:1 | | | |
|-----|---|----|-----|
| I | K | 30 | 100 |

Employee(ssn, ename, age, office_no)

↳ 한 번에 여러 개의 열(행)으로 office_no 있는 경우

영원 같은 개념적 office 할당은 4값

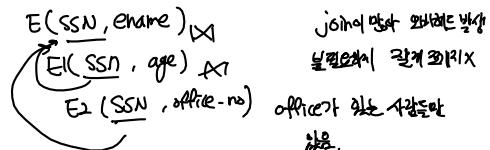
불가능한 경우 1이 문제

대부분의 Employee는 개별 office_no를 가지고 있지 않다. 위의 Employee를 아래와 같이 두 개로 분리하는 것이 좋다.

- Employee(ssn, ename, age)

→ 모든 행은 의미가 많아서 좋겠다.

Emp_Office(ssn, office_no)



가짜 투플 (spurious tuples)

- 관계 데이터베이스 설계를 잘못하게 되면, 조인 연산들이 틀린 결과를 생성할 수 있다.
- 조인 연산의 결과가 올바르기 위해서는, 릴레이션들이 “무손실 조인(lossless join)” 조건을 만족하도록 설계되어야 한다.

(a)

EMP_LOCS

| Ename | Plocation |
|-------|-----------|
| | |

P.K.



EMP_PROJ1

| Ssn | Pnumber | Hours | Pname | Plocation |
|-----|---------|-------|-------|-----------|
| | | | | |

P.K.

join attribute
Plocation.

아예 합계가 아니요
이상한 내용들이 들어온다.

Figure 15.5

Particularly poor design for the EMP_PROJ relation in Figure 15.3(b). (a) The two relation schemas EMP_LOCS and EMP_PROJ1. (b) The result of projecting the extension of EMP_PROJ from Figure 15.4 onto the relations EMP_LOCS and EMP_PROJ1.

(b)

EMP_LOCS $\leftarrow \pi_{...}(\text{EMP_PROJ})$



| Ename | Plocation |
|----------------------|-----------|
| Smith, John B. | Bellaire |
| Smith, John B. | Sugarland |
| Narayan, Ramesh K. | Houston |
| English, Joyce A. | Bellaire |
| English, Joyce A. | Sugarland |
| Wong, Franklin T. | Sugarland |
| Wong, Franklin T. | Houston |
| Wong, Franklin T. | Stafford |
| Zelaya, Alicia J. | Stafford |
| Jabbar, Ahmad V. | Stafford |
| Wallace, Jennifer S. | Stafford |
| Wallace, Jennifer S. | Houston |
| Borg, James E. | Houston |

EMP_PROJ1 $\leftarrow \pi_{...}(\text{EMP_PROJ})$

설계와 맞지 않아

| Ssn | Pnumber | Hours | Pname | Plocation |
|-----------|---------|-------|-----------------|-----------|
| 123456789 | 1 | 32.5 | ProductX | Bellaire |
| 123456789 | 2 | 7.5 | ProductY | Sugarland |
| 666884444 | 3 | 40.0 | ProductZ | Houston |
| 453453453 | 1 | 20.0 | ProductX | Bellaire |
| 453453453 | 2 | 20.0 | ProductY | Sugarland |
| 333445555 | 2 | 10.0 | ProductY | Sugarland |
| 333445555 | 3 | 10.0 | ProductZ | Houston |
| 333445555 | 10 | 10.0 | Computerization | Stafford |
| 333445555 | 20 | 10.0 | Reorganization | Houston |
| 999887777 | 30 | 30.0 | Newbenefits | Stafford |
| 999887777 | 10 | 10.0 | Computerization | Stafford |
| 987987987 | 10 | 35.0 | Computerization | Stafford |
| 987987987 | 30 | 5.0 | Newbenefits | Stafford |
| 987654321 | 30 | 20.0 | Newbenefits | Stafford |
| 987654321 | 20 | 15.0 | Reorganization | Houston |
| 888665555 | 20 | NULL | Reorganization | Houston |

따라서, join operation의 meaningful results를 보장하기 위해 "loseless join property" (join이, 소트는 전화 있는 투영)
* 사용한다.

| Ssn | Pnumber | Hours | Pname | Plocation | Ename |
|-----------|-----------|-------|-----------------|-----------------------------|-----------------------------|
| 123456789 | 1 | 32.5 | ProductX | Bellaire | Smith, John B. |
| * | 123456789 | 1 | 32.5 | ProductX | Bellaire English, Joyce A. |
| 123456789 | 2 | 7.5 | ProductY | Sugarland | Smith, John B. |
| * | 123456789 | 2 | 7.5 | ProductY | Sugarland English, Joyce A. |
| * | 123456789 | 2 | 7.5 | ProductY | Sugarland Wong, Franklin T. |
| 666884444 | 3 | 40.0 | ProductZ | Houston | Narayan, Ramesh K. |
| * | 666884444 | 3 | 40.0 | ProductZ | Houston Wong, Franklin T. |
| * | 453453453 | 1 | 20.0 | ProductX | Bellaire Smith, John B. |
| 453453453 | 1 | 20.0 | ProductX | Bellaire English, Joyce A. | |
| * | 453453453 | 2 | 20.0 | ProductY | Sugarland Smith, John B. |
| 453453453 | 2 | 20.0 | ProductY | Sugarland English, Joyce A. | |
| * | 453453453 | 2 | 20.0 | ProductY | Sugarland Wong, Franklin T. |
| * | 333445555 | 2 | 10.0 | ProductY | Sugarland Smith, John B. |
| * | 333445555 | 2 | 10.0 | ProductY | Sugarland English, Joyce A. |
| 333445555 | 2 | 10.0 | ProductY | Sugarland Wong, Franklin T. | |
| * | 333445555 | 3 | 10.0 | ProductZ | Houston Narayan, Ramesh K. |
| 333445555 | 3 | 10.0 | ProductZ | Houston Wong, Franklin T. | |
| 333445555 | 10 | 10.0 | Computerization | Stafford Wong, Franklin T. | |
| * | 333445555 | 20 | 10.0 | Reorganization | Houston Narayan, Ramesh K. |
| 333445555 | 20 | 10.0 | Reorganization | Houston Wong, Franklin T. | |

* * * EMP_PROJ1을 EMP_LOCS와 EMP_PROJ1로 병합

• EMP_LOCS * EMP_PROJ1 → 중복발생, 이중한 Tuples 발생 가능

"Inconsistency" . 두 relation의 JOIN이 incorrect original Information을 생성하기 때문이다.
• Location은 두 relation에서 PK, FK로 선택된다.
• 이는 같은 전형적 Attribute를 PK나, FK로 선택하는 경우이다

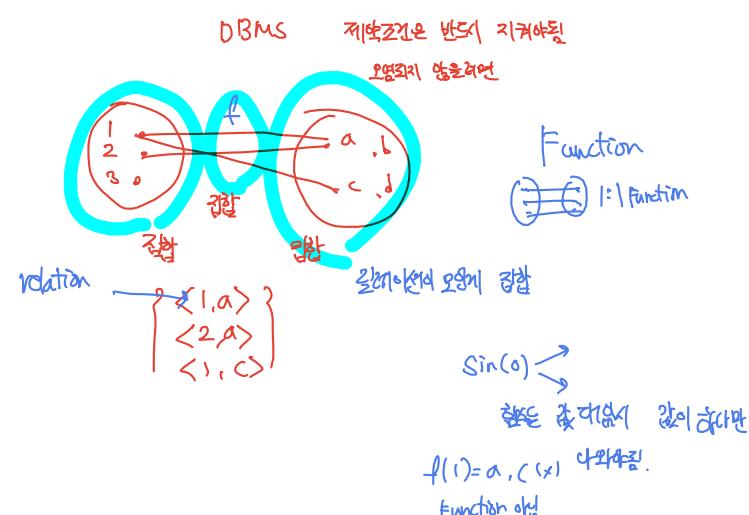
Figure 15.6

Result of applying NATURAL JOIN to the tuples above the dashed lines in EMP_PROJ1 and EMP_LOCS of Figure 15.5. Generated spurious tuples are marked by asterisks.

함수적 종속성

key or candidate key PK(X)

- 함수적 종속성(FD: functional dependency)은 좋은 릴레이션 설계의 정형적 기준으로 사용된다.
- FD와 키는 릴레이션의 정규형을 정의하기 위해 사용된다.
- FD는 데이터 애트리뷰트들의 의미와 애트리뷰트들 간의 상호 관계로부터 유도되는 제약조건(constraints)의 일종이다.
- 구성
 - 함수의 종속성(functional dependency)의 정의
 - 함수적 종속성의 추론 규칙
 - 함수적 종속성 집합의 동등성
 - 함수적 종속성의 최소집합



keys는 relations의 normal form을 정하는게 사용

X의 value가 Y의 "unique" value를 결정하면

"attribute X의 값은 attribute Y의 값을 functionally determines"

함수적 종속성의 정의

{ SSN, Pnumber }

• 함수적 종속성

- X와 Y를 임의의 애트리뷰트들의 집합이라고 할 때, X의 값이 Y의 값을 유일하게(unique) 결정한다면

"X는 Y를 함수적으로 결정한다(functionally determines)"라고 함 (X, Y는 R의 subset)

- X → Y로 표기하고, "Y는 X에 함수적으로 종속된다"라고 함 (값은 X의 값에만 관계)

- 함수적 종속성은 모든 릴레이션 인스턴스 r(R)에 대하여 성립해야 함

• 함수적 종속성의 검사 방법

- 릴레이션 인스턴스 r(R)에 속하는 어떠한 임의의 두 투플에 대해서도 속성들의 집합 X에 대해 동일한

값을 가질 때마다 Y에 대해서도 동일한 값을 가진다면 X → Y라는 함수적 종속성이 성립한다.

- 즉, r(R)에서의 임의의 두 투플 t_1 과 t_2 에 대해 $t_1[X] = t_2[X]$ 이면, $t_1[Y] = t_2[Y]$ 이다.

• FD는 특정 릴레이션 인스턴스보다는 실세계에서 존재하는 애트리뷰트들 사이의 제약조건으로부터 유도된다.

• FD 제약조건의 예제

- 주민등록번호는 사원의 이름을 결정한다.

◦ SSN → ENAME

- 프로젝트 번호는 프로젝트 이름과 위치를 결정한다.

◦ PNUMBER → {PNAME, PLOCATION}

- 사원의 주민등록번호와 프로젝트 번호는 그 사원이 일주일동안 그 프로젝트를 위해서 일하는 시간을 결정한다.

◦ {SSN, PNUMBER} → HOURS

- FD는 스키마 R에 있는 애트리뷰트들의 특성이며, 모든 릴레이션 인스턴스 r(R)에서 성립해야 하는 성질이다.

- K가 R의 키라면 K는 R의 모든 애트리뷰트들을 함수적으로 결정한다 → 키에 대해서는 동일한 값을 갖는 두개의 $(t_1[K] = t_2[K])$ 인 서로 다른 두개의 투플이 존재하지 않기 때문에). "distinct" tuple을 가질 수 없기 때문이다.

$X \rightarrow Y$ 라고 해서 반드시 $Y \rightarrow X$ 인 것은 아니라

함수적 종속성의 추론 규칙

- 설계자는 주어진(알려진) FD의 집합 F를 가지고, 추가로 성립하는 FD들을 추론할 수 있다.

• 암스트롱의 추론 규칙들

- A1. (재귀성 규칙) $Y \subseteq X$ 이면, $X \rightarrow Y$ 이다.
- A2. (부가성 규칙) $X \rightarrow Y$ 이면, $XZ \rightarrow YZ$ 이다. (표기: XZ 는 $X \cup Z$ 를 의미)
- A3. (이행성 규칙) $X \rightarrow Y$ 이고 $Y \rightarrow Z$ 이면, $X \rightarrow Z$ 이다.

- A1, A2, A3는 sound하고 complete 추론 규칙 집합을 형성한다.

• 추가적으로 유용한 추론 규칙들

- (분해 규칙) $X \rightarrow YZ$ 이면, $X \rightarrow Y$ 이고 $X \rightarrow Z$ 이다.
- (합집합 규칙) $X \rightarrow Y$ 이고 $X \rightarrow Z$ 이면, $X \rightarrow YZ$ 이다.
- (의사이행성 규칙) $X \rightarrow Y$ 이고 $WY \rightarrow Z$ 이면, $WX \rightarrow Z$ 이다.

- 위의 세 규칙 뿐 아니라 다른 추론 규칙들도 A1, A2, A3으로부터 추론 가능하다 (완전성 특성).

• FD의 집합 F의 폐포(closure) : F^+

- F로부터 추론할 수 있는 모든 가능한 함수적 종속성들의 집합

• F 하에서 속성 집합 X의 폐포(closure of X under F) : X^+

- 함수적 종속성 집합 F를 사용하여 X에 의해 함수적으로 결정되는 모든 애트리뷰트들의 집합

$$X^+ \text{ under } F = \{X, Y, W, Z\}$$

이행성 규칙

예제: $X \rightarrow Y$ $\rightarrow X \rightarrow W$ $\rightarrow X \rightarrow Z$

여기서 $X \rightarrow W$ 이 있으므로 $X \rightarrow Z$ 를 증명할 수 있다.

이행성 규칙

주어진 relation instance에 대해서, FD는

특정 attributes 사이에 조건을 수도 있다:

in CAR, ?State, Driver_license_number? → ?SSN?

dependencies의 위법을 보는 tuples로 인해 특정

$A \rightarrow B$
 $B \rightarrow C$

Sech, IPN → 강의 태그

① $|A \rightarrow B, B \rightarrow C|$ → 강의실 → 강의실 태그.
두번만 같은 태그로
올릴 수 있다.

② $|A \rightarrow B, B \rightarrow C, A \rightarrow C|$

in Project, ?Place? → ?Pnumber?
동일한 Place와 Project가
존재할 수도 있기 때문이다.

$R(A, B, C)$
 $P: A \rightarrow D, B \rightarrow C$

$F^+ = \{A \rightarrow B, B \rightarrow C, C \rightarrow C\}$
 $AB \rightarrow A, AB \rightarrow B, A \rightarrow C$

| | | |
|--------|---------|--------|
| A | B | C |
| A^+ | B^+ | C^+ |
| C^+ | AB^+ | |
| AB^+ | BC^+ | AC^+ |
| BC^+ | ABC^+ | |

알고리즘 : F하의 X의 폐포 X⁺를 구하는 알고리즘

프로그램의 앞에서 결정한다.

이 알고리즘은 complete 하다.

일부러 Input 시킬 필요 없어

$$X^+ := X \text{ 재귀성 규칙}$$

repeat

$$oldX^+ := X^+ \text{ 지금까지 구한 } X^+ \text{을 } oldX^+ \text{에 Save}$$

for each functional dependency $Y \rightarrow Z$ in F do 5번 반복문에
if $Y \subseteq X^+$ then $X^+ := X^+ \cup Z$ 돌아갔다.

until ($oldX^+ = X^+$)

파이썬에 Set 이용해 고대로 쪽으면 됨

$$FD = \{AB \rightarrow C, AC \rightarrow AD, CD \rightarrow F, A \rightarrow E, D \rightarrow E\}$$

$$Y \rightarrow Z$$

$$AB^+ \text{ under } F = ?$$

$$Y \rightarrow Z$$

$$X^+ \rightarrow Y \text{ 재귀성 (복잡성을 줄여야 함)}$$

$$\textcircled{1} \quad X^+ = ?A, B, C, A, D, F, E, ?$$

$$oldX^+ = ?A, B ?$$

$$X^+ \neq oldX^+ = ?A, B$$

$$X^+ \rightarrow oldX^+ = ?A, B, C, D, E, F ?$$

$$X^+ = ?A, B, C, D, E, F ?$$

$$X^+ = oldX^+ \text{ closure}$$

$AB^+ = A$ 가 수퍼키인 사실은 이전부터 알았음.
 $R(A, B, C, D, E, F)$ 수퍼키

Example :

$F =$

$$SSN \rightarrow ENAME,$$

$$PNUMBER \rightarrow PNAME, PLOCATION,$$

$$SSN, PNUMBER \rightarrow HOURS$$

알고리즘을 사용하여 F하에서 다음과 같은 폐포 집합들을 구할 수 있다.

- $SSN^+ = SSN, ENAME$

- $PNUMBER^+ = PNUMBER, PNAME, PLOCATION$

- $SSN, PNUMBER^+ = SSN, PNUMBER, ENAME, PNAME, PLOCATION, HOURS$

$$\textcircled{2} \quad AB^+ = ABCDEF$$

$$AD \not\subseteq A$$

$$A^+ = ?A, E ?$$

$$\textcircled{3} \quad B^+ = ?B, E ?$$

$$AB^+ = ABCDEF \rightarrow S.K$$

$$A^+ = AE \rightarrow C.K$$

$$B^+ = BE \rightarrow C.K$$

$$F^+ = g^+ \text{로부터 유도되어지는 것들에 대해서}$$

둘째하고 핫스는 것.

둘개가 의미하는 바가 같다 = 동등화하는 것을

제거하는 것을.

$H = ?A \rightarrow B, B \rightarrow C ?$ 를 유지하고 싶음. 주제로 검사해야 하는 계약조건에서 계산하기 편하도록 하는 것을 최소화하는 것을 찾고 싶음.

함수적 종속성 집합의 동등성

• 정의: cover

▪ G의 모든 FD가 F로부터 추론될 수 있다면(즉, $G^+ \subseteq F^+$) “F가 G를 덮는다(cover)”라고 말한다. minimal을 찾고 싶음.

• 두 FD 집합의 동등성

▪ FD의 집합 F와 G에 대하여, F의 모든 FD가 G로부터 추론될 수 있고,

• G의 모든 FD가 F로부터 추론될 수 있으면 “F와 G는 동등하다(equivalent)”라고 한다

▪ F와 G가 다르더라도 $F^+ = G^+$ 이면 F와 G는 동등하다.

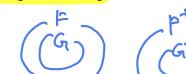
▪ F가 G를 덮고 G가 F를 덮으면 F와 G는 동등하다.

• FD 집합의 동등성을 검사하는 알고리즘이 존재함

$$F = ?A \rightarrow B, B \rightarrow C, A \rightarrow C, A \rightarrow A ?$$

$$g = ?A \rightarrow B, B \rightarrow C, AB \rightarrow A ?$$

$$H = ?A \rightarrow B, B \rightarrow C ?$$



From F, G를 cover하는지 증명하려면
① $G^+ \subseteq F^+$
② $F^+ \subseteq G^+$

if F cover G:
for each FD: $X \rightarrow Y$ in G
 Y must be in X^+ under F

$$\text{prove } F = ?A \rightarrow C, AC \rightarrow D, E \rightarrow AB, E \rightarrow M ? \quad \textcircled{A} \quad G^+ \subseteq F^+ \quad \textcircled{1} \quad A \rightarrow CD \text{ under } F$$

$A^+ \text{를 } F \text{하에서 구해보니 } CD \text{가 안나오면 됨.}$

$$A^+ \text{ under } F = ?A, C, D ?$$

∴ True

$$\textcircled{2} \quad E \rightarrow AH \text{ under } F$$

$$E^+ \text{ under } F = ?E, A, D, H, C ?$$

: True

증명됨

$$\textcircled{B} \quad F^+ \subseteq G^+ \quad \textcircled{3} \quad A \rightarrow C \text{ under } G \quad \text{True}$$

$$\textcircled{4} \quad AC \rightarrow D \text{ `` True''}$$

$$\textcircled{5} \quad E \rightarrow AD \text{ `` True''}$$

증명됨

$$\textcircled{6} \quad E \rightarrow H \text{ `` True''}$$

유지보수화 할. → G를 유지하자 보기 때문에

함수적 종속성의 최소집합

- 함수적 종속성들의 집합 F의 최소(minimal)
 - 다음 세 조건을 만족하는 FD 집합을 최소라고 함 $X \rightarrow Y$
 - F의 모든 함수적 종속성들의 오른쪽면 애트리뷰트가 하나이다.
 - F로부터 어떤 함수적 종속성을 제거했을 때, F와 동등한 함수적 종속성들의 집합이 될 수 없다.
 - F에서 X → A를 X의 진부분집합 Y에 대하여 Y → A로 교체했을 때, F와 동등한 함수적 종속성들의 집합이 될 수 없다.
- 함수적 종속성들의 집합 F의 최소 덮개(minimal cover)는 F와 동등한 함수적 종속성들의 최소 집합 F_{min} 을 의미함
 - 함수적 종속성들의 최소 덮개는 여러 개 존재할 수 있다.
 - 또한, 임의의 함수적 종속성들의 집합 F에 대해 알고리즘을 사용하여 적어도 하나의 최소 덮개를 구할 수 있다

$E : \{B \rightarrow A, D \rightarrow A, AB \rightarrow D\}$
E'에서 유도되어질수 있다.
 $E' : \{B \rightarrow A, D \rightarrow A, A \rightarrow D, D \rightarrow A\}$
 $AB^+ \text{ under } E' = ABD$
E'에서 유도되어질수 있다.
 $A^+ \text{ under } E = A$
 $B^+ \text{ under } E = BAD$
E'에서 유도되어질수 있다.
 $E'' : \{D \rightarrow A, B \rightarrow D\}$
 $E'' \text{의 } B \rightarrow D \text{은 } D \rightarrow A \text{이다.}$
 $B^+ \text{ under } E'' = B, D, A$
 $B \rightarrow BDA \rightarrow A$
E'에서 유도되어질수 있다.
 $F = \{X \rightarrow Y, V \rightarrow V\}$
 $F' = \{U \rightarrow V\}$

기본 키를 기반으로 한 정규형

- 정규화 소개
- 제1 정규형(First Normal Form ; 1NF)
- 제2 정규형(Second Normal Form ; 2NF)
- 제3 정규형(Third Normal Form ; 3NF)

정규화 소개

- 정규화(normalization)
 - 불만족스러운 “나쁜” 릴레이션의 애트리뷰트들을 나누어서 더 작은 “좋은” 릴레이션으로 분해하는 과정
- 정규형(normal form)
 - 특정 조건을 만족하는 릴레이션 스키마의 형태
- 제 2 정규형, 제 3 정규형, BCNF
 - 릴레이션 스키마의 FD와 키에 기반하여 정의됨
- 좋은 릴레이션 설계를 위해서는 정규형 이외에도 추가 특성이 필요함 (즉, 무손실 조인과 종속성 보존; 다음 장 참고)

제 1 정규형(1NF)

atomic ×

- 제 1 정규형
 - 복합 애트리뷰트(composite attribute), 다치 애트리뷰트(multivalue attribute), 그리고 중첩 릴레이션(nested relation) 등 비원자적(non-atomic) 애트리뷰트들을 허용하지 않은 릴레이션의 형태
- 제 1 정규형은 릴레이션 정의의 일부분을 이루고 있음

(a)

DEPARTMENT

| Dname | Dnumber | Dmgr_ssn | Dlocations |
|-------|---------|----------|------------|
| | | | |

(b)

DEPARTMENT

| Dname | Dnumber | Dmgr_ssn | Dlocations |
|----------------|---------|-----------|--------------------------------|
| Research | 5 | 333445555 | {Bellaire, Sugarland, Houston} |
| Administration | 4 | 987654321 | {Stafford} |
| Headquarters | 1 | 888665555 | {Houston} |

1NF

D1(Dnumber, Dlocation)

D2(Dname, Dnumber, Dmgr_ssn)

(c)

DEPARTMENT

| Dname | Dnumber | Dmgr_ssn | Dlocation |
|----------------|---------|-----------|-----------|
| Research | 5 | 333445555 | Bellaire |
| Research | 5 | 333445555 | Sugarland |
| Research | 5 | 333445555 | Houston |
| Administration | 4 | 987654321 | Stafford |
| Headquarters | 1 | 888665555 | Houston |

Figure 15.9

Normalization into 1NF. (a) A relation schema that is not in 1NF. (b) Sample state of relation DEPARTMENT. (c) 1NF version of the same relation with redundancy.

| dnumber | dlocation |
|---------|-----------|
| 5 | bellaire |
| 5 | H |
| 4 | S |
| 1 | H |

X

| dname | dnumber | dmgr_ssn |
|-------|---------|----------|
| R | 5 | 3 |
| A | 4 | 9 |
| H | 1 | 8 |

(a)

| EMP_PROJ | | Projs | |
|----------|-------|---------|-------|
| Ssn | Ename | Pnumber | Hours |

(b)

| EMP_PROJ | | | |
|-----------|----------------------|---------|-------|
| Ssn | Ename | Pnumber | Hours |
| 123456789 | Smith, John B. | 1 | 32.5 |
| | | 2 | 7.5 |
| 666884444 | Narayan, Ramesh K. | 3 | 40.0 |
| 453453453 | English, Joyce A. | 1 | 20.0 |
| | | 2 | 20.0 |
| 333445555 | Wong, Franklin T. | 2 | 10.0 |
| | | 3 | 10.0 |
| | | 10 | 10.0 |
| | | 20 | 10.0 |
| 999887777 | Zelaya, Alicia J. | 30 | 30.0 |
| | | 10 | 10.0 |
| 987987987 | Jabbar, Ahmad V. | 10 | 35.0 |
| | | 30 | 5.0 |
| 987654321 | Wallace, Jennifer S. | 30 | 20.0 |
| | | 20 | 15.0 |
| 888665555 | Borg, James E. | 20 | NULL |

Figure 15.10

Normalizing nested relations into 1NF. (a) Schema of the EMP_PROJ relation with a *nested relation* attribute PROJS. (b) Sample extension of the EMP_PROJ relation showing nested relations within each tuple. (c) Decomposition of EMP_PROJ into relations EMP_PROJ1 and EMP_PROJ2 by propagating the primary key.

(c)

| EMP_PROJ1 | | |
|------------------|---------|-------|
| Ssn | Ename | |
| EMP_PROJ2 | | |
| Ssn | Pnumber | Hours |

1NF Nested relation 복합�性 attribute

$D^+ \subset E$

제 2 정규형(2NF)

- 제 2 정규형은 FD와 기본키 개념을 이용한다.

- 정의:

① 어떤 애트리뷰트를 참조하는지 알 수가 있다면 그가 primary key

- 기본 애트리뷰트(prime attribute): 기본키 K의 멤버인 애트리뷰트

- 완전 함수적 종속성(full functional dependency): FD $Y \rightarrow Z$ 에서 Y의 어떤 애트리뷰트라도 제거하면 더이상 성립하지 않는 경우

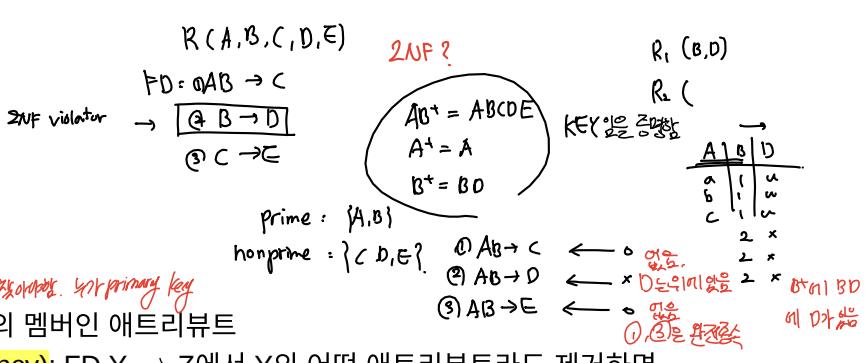
- 예제:

- {SSN, PNUMBER} \rightarrow HOURS는 SSN \rightarrow HOURS와 PNUMBER \rightarrow HOURS가 성립하지 않기 때문에 완전 함수적 종속성이다.

- {SSN, PNUMBER} \rightarrow ENAME은 SSN \rightarrow ENAME이 성립하기 때문에 완전 함수적 종속성이 아니다 (이는 부분 함수 종속성(partial function dependency)이라고 부름).

- 릴레이션 스키마 R의 모든 비기본(non-prime) 애트리뷰트들이 기본키에 대해서 완전 함수적 종속이면, R은 제 2 정규형(2NF)에 속한다.

- R은 제 2 정규형 정규화 과정에 의해서 항상 제 2 정규형 릴레이션으로 분해될 수 있다.



SCR $S \rightarrow Y$ Y의 전부분집합을 제거하면 결점x

문맥을 빼면 상관없으면

2NFx

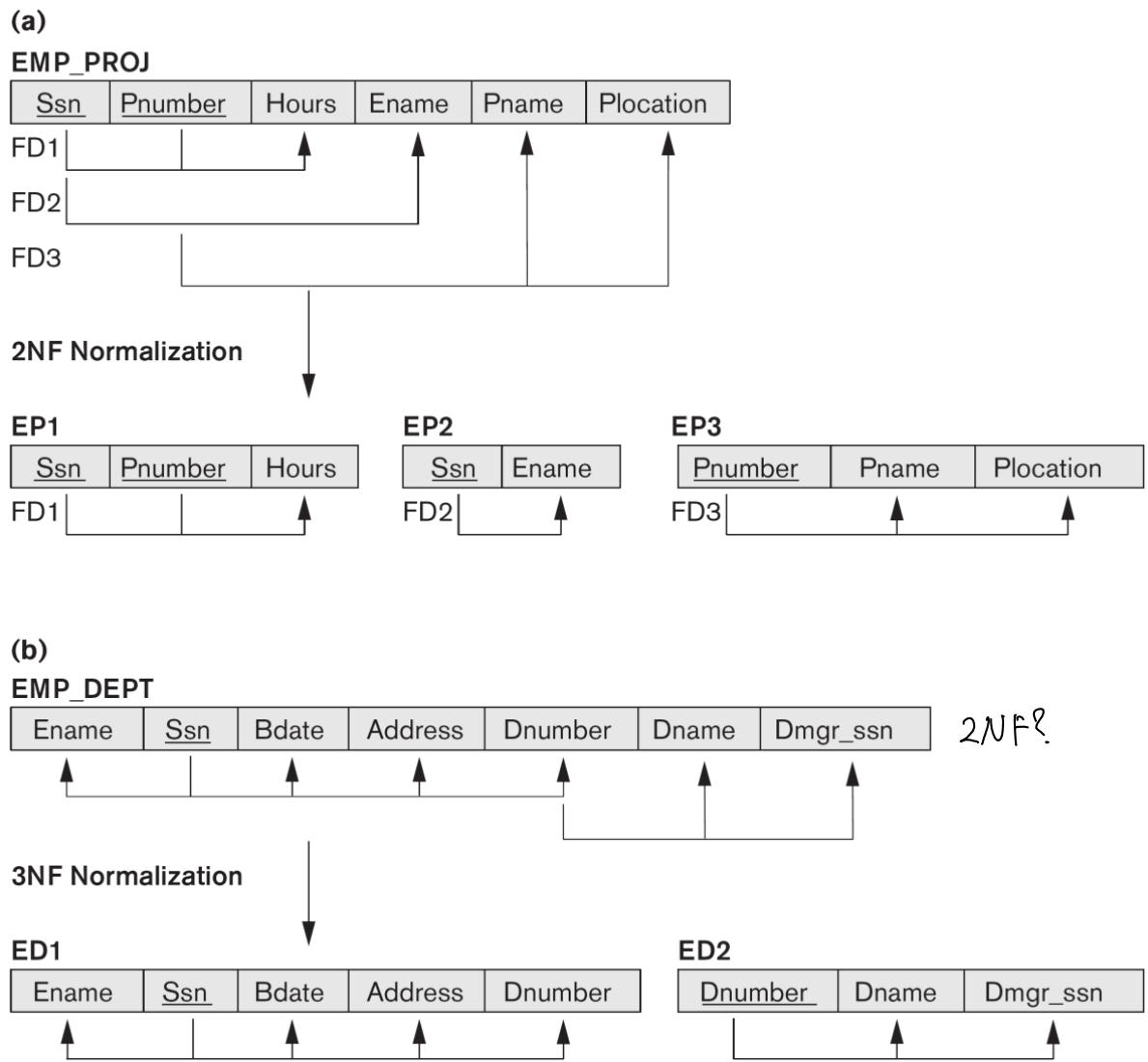


Figure 15.11

Normalizing into 2NF and 3NF. (a) Normalizing EMP_PROJ into 2NF relations. (b) Normalizing EMP_DEPT into 3NF relations.

제 3 정규형(3NF)



- 정의
 - 이행적 함수적 종속성(transitive functional dependency): 두 FD $Y \rightarrow X$ 와 $X \rightarrow Z$ 에 의해서 추론될 수 있는 FD $Y \rightarrow Z$
- 예제
 - SSN → DMGRSSN은 SSN → DNUMBER과 DNUMBER → DMGRSSN이 성립하기 때문에 이 행적 함수적 종속성이다.
 - SSN → ENAME는 SSN → X이고 X → ENAME인 애트리뷰트 집합 X가 존재하지 않기 때문에 이 행적 종속성이 아니다.
- 릴레이션 스키마 R이 제 2 정규형을 갖고 R의 어떤 비기본 애트리뷰트도 기본키에 대해서 이행적으로 종속되지 않으면 R은 제 3 정규형을 갖는다고 한다.
- R은 제 3 정규형 정규화 과정에 의해서 항상 제 3 정규형 릴레이션으로 분해될 수 있다

제 2 정규형과 제 3 정규형의 일반적인 정의

- 여기서부터는 여러개의 후보 키를 가진 릴레이션들을 고려한다.
- 릴레이션 스키마 R의 모든 **비기본** 애트리뷰트 A가 R의 **모든 후보키에 완전 함수적 종속**이면 R은 제 2 정규형(2NF)을 갖는다고 한다.
- 정의:
 - 기본 애트리뷰트(prime attribute):** 임의의 후보키 K의 멤버인 애트리뷰트
 - 릴레이션 스키마 R의 수퍼키(superkey):** R의 후보키를 포함한 R의 애트리뷰트들의 집합 S
 - 릴레이션 스키마 R의 FD $X \rightarrow A$** 가 성립할 때마다
 - (a) X 가 R의 수퍼키 이거나
 - (b) A가 R의 기본 애트리뷰트이면 R은 제 3 정규형(3NF)을 갖는다고 한다.
- Boyce-Codd 정규형은 위의 조건 중 (b)의 경우를 허락치 않는 정규형을 의미한다.
반드시 만족해야 함.

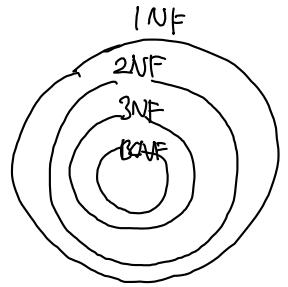
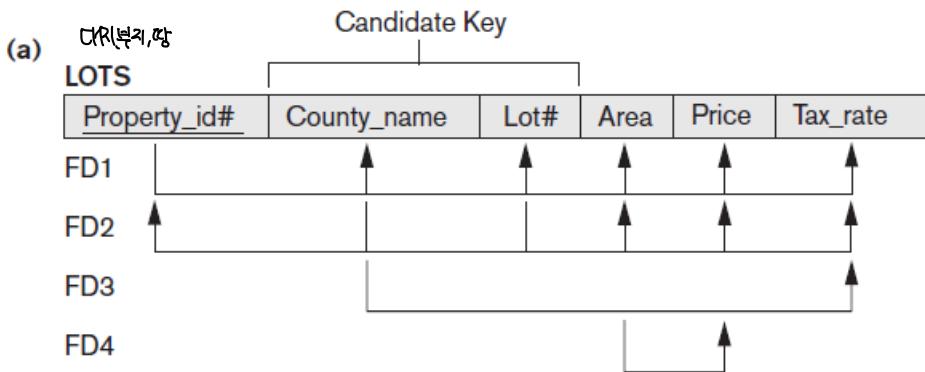


Figure 15.12

Normalization into 2NF and 3NF. (a) The LOTS relation with its functional dependencies FD1 through FD4. (b) Decomposing into the 2NF relations LOTS1 and LOTS2. (c) Decomposing LOTS1 into the 3NF relations LOTS1A and LOTS1B. (d) Summary of the progressive normalization of LOTS.



CK: {Property_id#, County_name, Lot#}

Prime: {pid, cn, lot#} $pid \rightarrow TR$

2NF?

CN^+

Example.

① $R(A, B, C, D)$

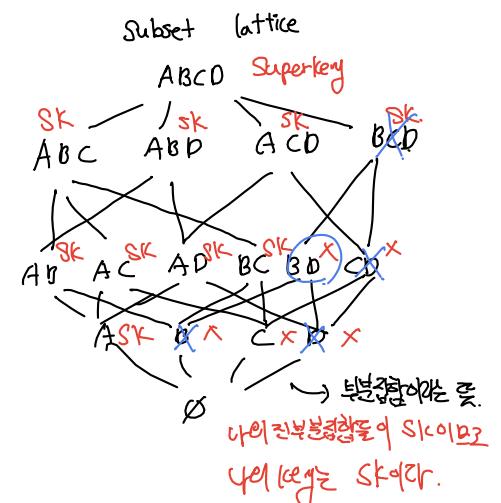
FD: $A \rightarrow B$

$A \rightarrow C$

$A \rightarrow D$

$BC \rightarrow A$

CK?



(b)

LOTS1

| <u>Property_id#</u> | County_name | Lot# | Area | Price |
|---------------------|-------------|------|------|-------|
| FD1 | | | | |
| FD2 | | | | |
| FD4 | | | | |

LOTS2

| County_name | Tax_rate |
|-------------|----------|
| FD3 | |

(c)

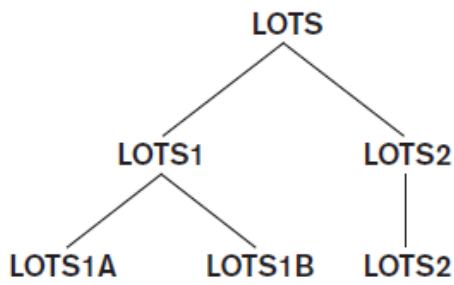
LOTS1A

| <u>Property_id#</u> | County_name | Lot# | Area |
|---------------------|-------------|------|------|
| FD1 | | | |
| FD2 | | | |

LOTS1B

| Area | Price |
|------|-------|
| FD4 | |

(d)



1NF

2NF

3NF

BCNF (Boyce-Codd Normal Form)

- 릴레이션 스키마 R에서 성립하는 임의의 FD $X \rightarrow A$ 에서 X가 R의 수퍼키이면 R은 Boyce-Codd 정규형 (BCNF)을 갖는다고 한다.
- 각 정규형은 그의 선행 정규형보다 더 엄격한 조건을 갖는다. 즉,
 - 모든 제 2 정규형 릴레이션은 제 1 정규형을 갖는다.
 - 모든 제 3 정규형 릴레이션은 제 2 정규형을 갖는다.
 - 모든 BCNF 릴레이션은 제 3 정규형을 갖는다.
- 제 3 정규형에는 속하나 BCNF에는 속하지 않는 릴레이션이 존재한다.
- 관계 데이터베이스 설계의 목표는 각 릴레이션이 BCNF(또는 제 3 정규형)를 갖게 하는 것이다.
- “좋은” 관계형 데이터베이스의 릴레이션을 설계하기 위해서는 추가적인 특성이 만족되어야 한다.
 - 무손실 조인(lossless join) 특성
 - 종속성 보존(dependency preservation) 특성

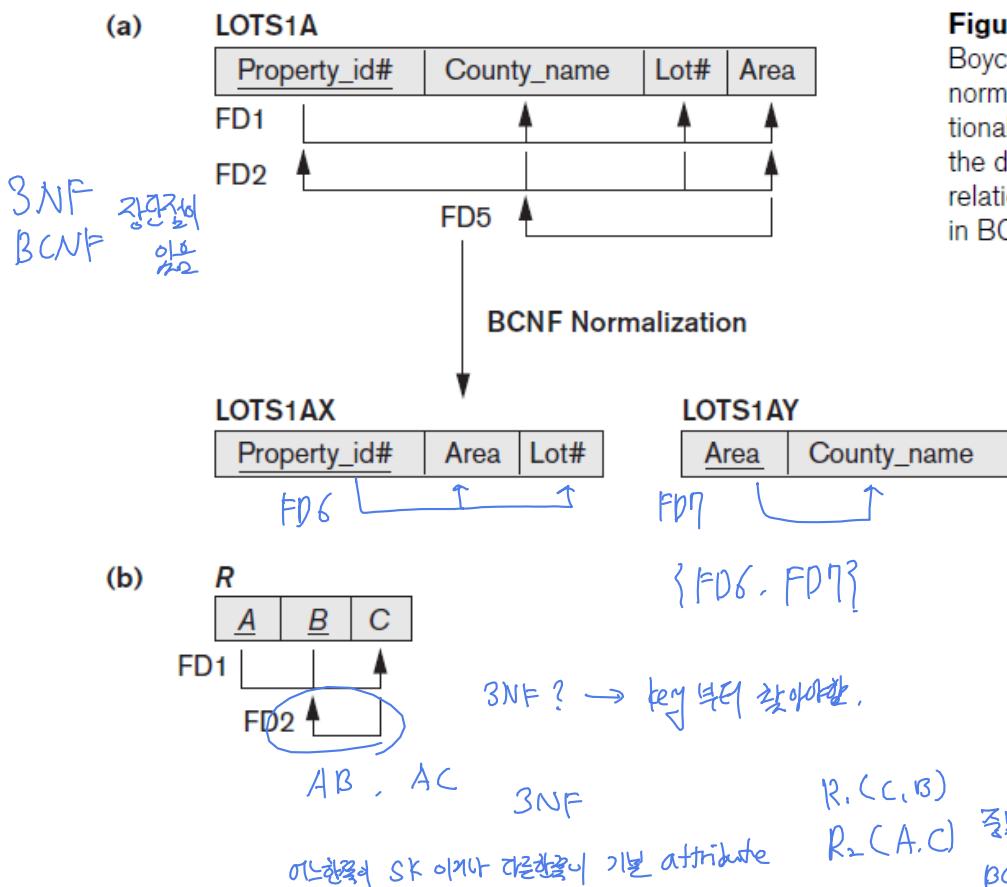


Figure 15.13

Boyce-Codd normal form. (a) BCNF normalization of LOTS1A with the functional dependency FD2 being lost in the decomposition. (b) A schematic relation with FDs; it is in 3NF, but not in BCNF.

Relation in 3NF but not BCNF

Figure 15.14

A relation TEACH that is in 3NF but not BCNF.

TEACH

| Student | Course | Instructor |
|---------|-------------------|------------|
| Narayan | Database | Mark |
| Smith | Database | Navathe |
| Smith | Operating Systems | Ammar |
| Smith | Theory | Schulman |
| Wallace | Database | Mark |
| Wallace | Operating Systems | Ahamad |
| Wong | Database | Omiecinski |
| Zelaya | Database | Navathe |
| Narayan | Operating Systems | Ammar |

FD:

Student, Course → Instructor

Instructor → Course

기본적임 학수과 종속성 찾기

Exercise 1

Consider a relation $R(A, B, C, D)$, with FDs $AB \rightarrow C$, $BC \rightarrow D$, $CD \rightarrow A$.

- (a) Find the closure of AB .
- (b) Is R a good schema?
- (c) If we decompose R as $R1(A,B,C)$ and $R2(A,C,D)$. Is it a good decomposition?

Exercise 2

- Consider relation $R(A,B,C,D,E)$ with the following functional dependencies: $AB \rightarrow C$, $D \rightarrow E$, $DE \rightarrow B$.
- Is R in BCNF? If not, decompose R into a collection of BCNF relations.

Exercise 3

- “Any two-attribute relation is in BCNF.” Is it correct?

Exercise 4

Compute the closure of the following set F of functional dependencies for relation schema $R = \{A, B, C, D, E\}$.

$A \rightarrow BC$
 $CD \rightarrow E$
 $B \rightarrow D$
 $E \rightarrow A$

List the candidate keys for R .

Exercise 5

Consider a relation $R(A,B,C,D,E)$ with the following dependencies:

$\{AB \rightarrow C, CD \rightarrow E, DE \rightarrow B\}$

List all candidate keys.

Exercise 6

$R(A,B,C,D)$ and FDs $\{AB \rightarrow C, C \rightarrow D, D \rightarrow A\}$.

- (1) List all nontrivial FDs that can be inferred from the given FDs.
- (2) Find all candidate keys.
- (3) Find all BCNF violations.
- (4) Decompose R into relations in BCNF.
- (5) What FDs are not preserved by BCNF.