CIS560

Obtaining a Good Database Design – Part 2

1

What's next?

- We briefly discussed the 3NF and BCNF.
- They are defined using:
  - Functional Dependencies
  - Keys
- We defined functional dependencies.
- We defined closures and how they help us find all functional dependencies.
- Now let's review keys.

2

Superkeys and Keys

- A **superkey** is a set of attributes A_1, ..., A_n s.t. for any other attribute B, we have A_1, ..., $A_n \rightarrow B$
- A **key** is a minimal superkey
  - A set of attributes which is a superkey
  - And for which no subset is a superkey

Computing Keys

- Compute X^+ for all sets X
- If X^+ = all attributes, then X is a superkey
- We only want the keys *(minimal superkeys)*

Example - What are the key(s)?

Enrollment(student, address, course, room, time)

student → address
room, time → course
student, course → room, time

HINT: You can have more than one key.

Keys: {student, room, time}, {student, course}

How do we use keys to eliminate anomalies?

Each attribute must provide a fact about the key, the whole key, and nothing but the key.

Chris Date's adaptation to William Kent's summary

Boyce-Codd Normal Form

- A relation R is in BCNF if and only if for every functional dependency X → A:
  - X → A is a trivial functional dependency
    or
  - X is a superkey for R

- Equivalently: ∀ X, where X is a set of attributes, either (X^+ = X) or (X^+ = all attributes)

Example

Name	ID	Phone	Department
Fred	123	206-555-1234	CIS
Fred	123	206-555-6543	CIS
Joe	987	908-555-2121	Math
Joe	987	206-151-7839	Math

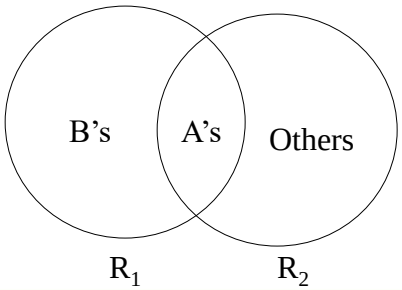
ID → Name, Department

What is the key?
{ID, Phone}

Hence ID → Name, Department
is a "bad" dependency

BCNF Decomposition – Using FDs

repeat
choose $A_1, \ldots, A_m \rightarrow B_1, \ldots, B_n$ that violates BNCF (bad FD)
split R into $R_1(A_1, \ldots, A_m, B_1, \ldots, B_n)$ and $R_2(A_1, \ldots, A_m, [\text{others}])$
continue with both R_1 and R_2
until no more violations



A two-attribute relation is always in BCNF.

Example

Name	ID	Phone	Department
Fred	123	206-555-1234	CIS
Fred	123	206-555-6543	CIS
Joe	987	908-555-2121	Math
Joe	987	206-151-7839	Math

ID → Name, Department is a bad functional dependency

Relation Decomposition

Name	ID	Department	Phone
Fred	123	CIS	206-555-1234
Fred	123	CIS	206-555-6543
Joe	987	Math	908-555-2121
Joe	987	Math	206-151-7839

Relation Decomposition

Name	ID	Department
Fred	123	CIS
Joe	987	Math

ID	Phone
123	206-555-1234
123	206-555-6543
987	908-555-2121
987	206-151-7839

BCNF Decomposition – Using Closures

repeat
find X s.t.: $X \neq X^+ \neq$ [all attributes]

if (not found) **then** "R is in BCNF"

let Y = [all attributes] - X^+
decompose R into R1(X^+) and R2($X \cup Y$)
continue with both R1 and R2
until no X is found

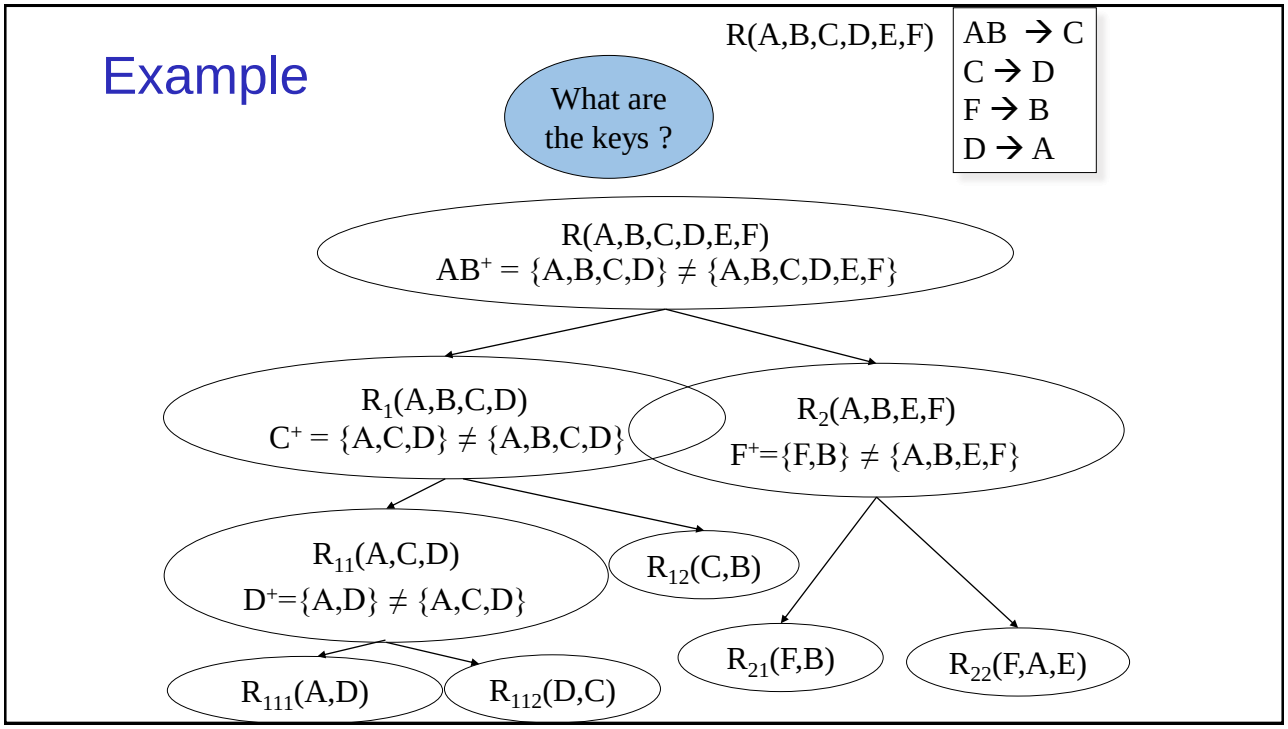
Example BCNF Decomposition – Using Closures

Student(name, ID, age, hairColor, phoneNumber)
ID → name, age
age → hairColor

Iteration 1: Student X = ID
ID^+ = {ID, name, age, hairColor}
Decompose into: Student1(ID, name, age, hairColor)
Phone(ID, phoneNumber)

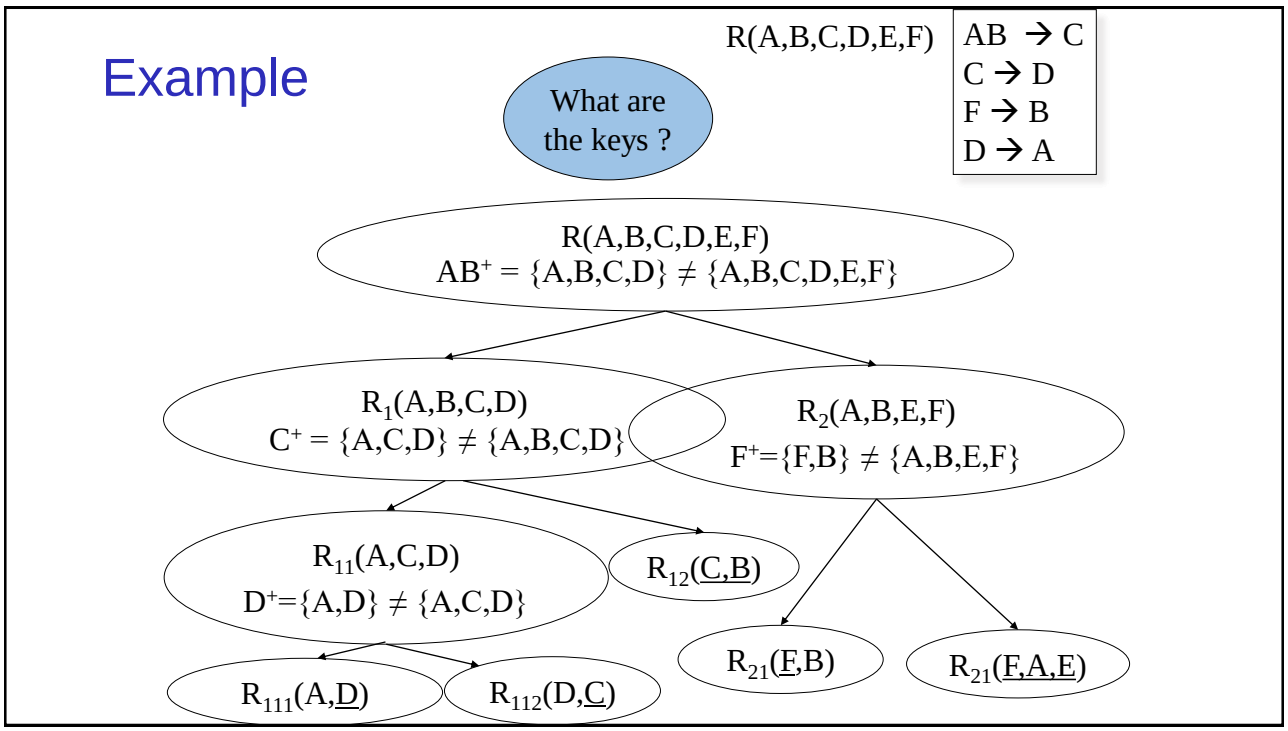
Iteration 2: Student1 X = age
age^+ = {age, hairColor}
Decompose: Hair(age, hairColor)
Student2(ID, name, age)

Example



15

Example



16

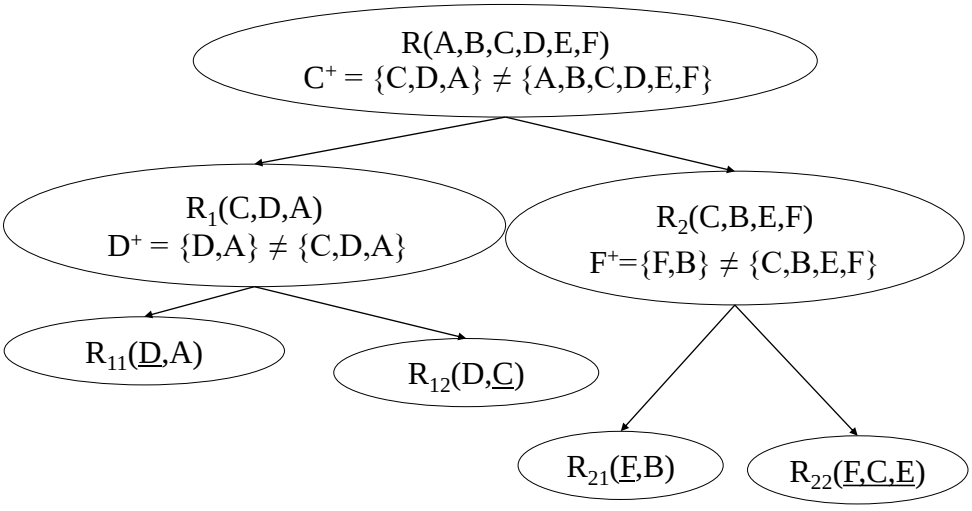
Is the resulting BCNF schema unique?

17

Example – Solution 2

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

AB → C
C → D
F → B
D → A



18

Which solution is better?

$R_{111}(A,\underline{D}), R_{112}(D,\underline{C}), R_{12}(\underline{C},\underline{B}), R_{21}(\underline{F},B), R_{22}(\underline{F},\underline{A},\underline{E})$

OR

$R_{11}(A,\underline{D}), R_{12}(D,\underline{C}), R_{21}(\underline{E},B), R_{22}(\underline{F},\underline{C},\underline{E})$

Which solution is better?

- From the theoretical point of view…
  Both solutions are good.
- From a practical point of view…
  It depends.
- Look at the common ways they are queried, for example.