Lossless Join Decomposition Tutorial

SWEN304/SWEN439

Lecturer: Dr Hui Ma

Engineering and Computer Science

- Lossless join decomposition

- Exercises

  - 3NF decomposition

  - BCNF decomposition

FDs and a Relation Schema Key

- Each relation schema **key** is the consequence of a functional dependency from F^+

- Let $R(A_1, \ldots, A_n)$ be a relation schema and F the set of functional dependencies in R

- Set of attributes $X \subseteq R$ is a relation schema **key** if

1° $X \rightarrow R \in F^+$ (or $X^+ = R$)

2° $(\forall Y \subset X)(Y \rightarrow R \notin F^+)$

- Not null condition still applies to X

- A **prime** attribute is a relation schema attribute that belongs to any of the keys

- Primary key is one of the keys

Lossless Join Decomposition

- A decomposition D = $\{R_1, R_2, ..., R_m\}$ of a relation R has the **lossless (nonadditive) join** property wrt the set of dependencies F on R if, for every relation $r(R)$ that satisfies F,

$$* (\pi_{R1}r(R), ..., \pi_{Rm}r(R)) = r(R)$$

where * is the natural join of all the relations in D.

- It is proven in the theory of the relational data model that the decomposition of a relation schema R onto R_1 and R_2 is *lossless (non-additive)* if the intersection $R_1 \cap R_2$ contains a **key** of R_1 or a key of R_2

Example 1: Checking Losslessness of D (1)

- Given a set of relation schemas:

D = {N_1({A, B }, {A}), N_2({B, D }, {B }), N_3({C, B }, {C })}

- How to check whether the whole set of relation schemas represents a lossless join decomposition of the (supposed) universal relation schema?

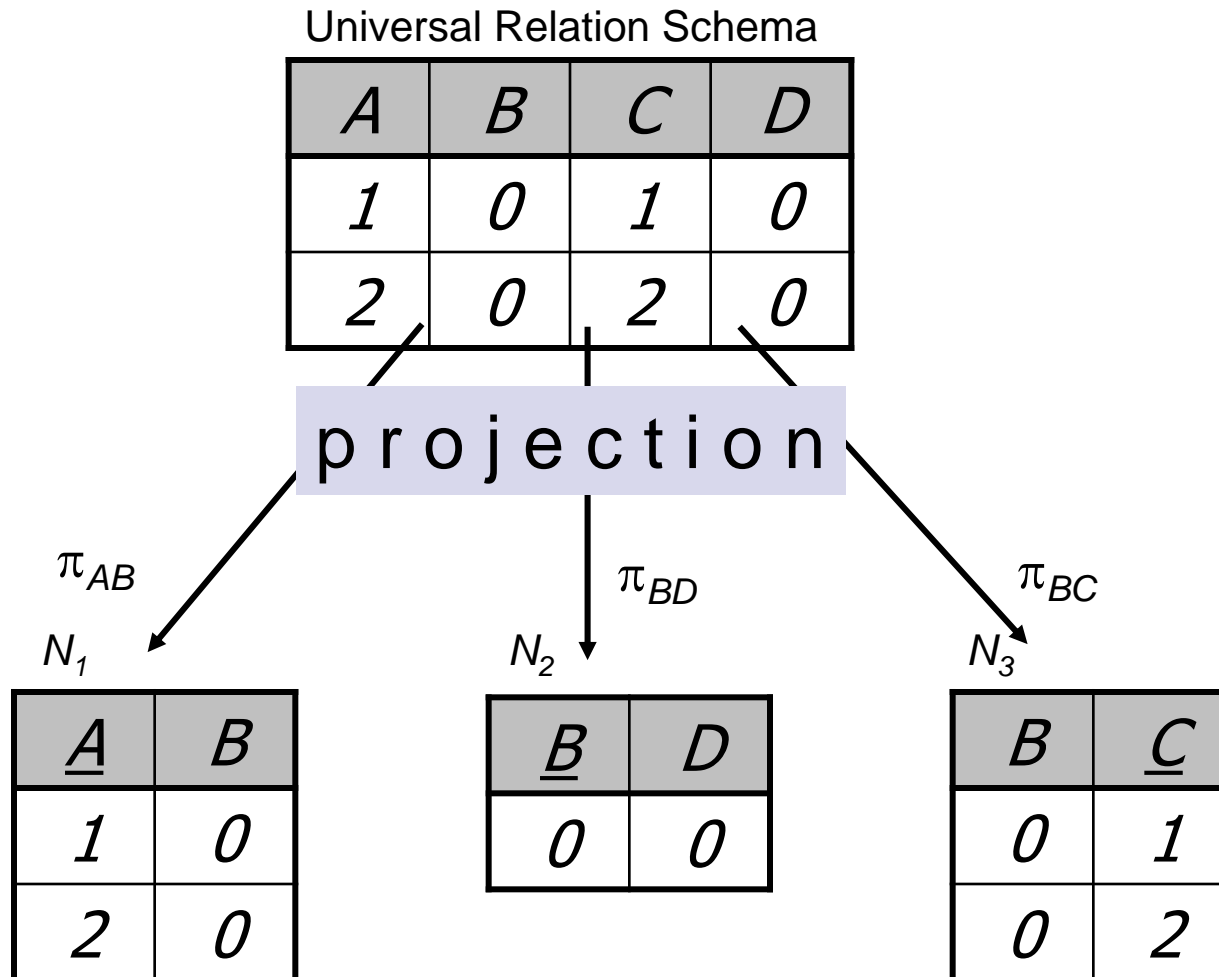# Example 1: Checking Losslessness of D (2)

- A naïve and generally **wrong** approach:

  - Perform a pair wise checking of the relation schemas

    - for each relation schema you find another one such that the intersection of the two schemas is a schema key of one of the schema

- So, according to that approach:

  - $\{A, B\} \cap \{B, D\} = \{B\}$, and B is the key of N_2

  - $\{B, C\} \cap \{B, D\} = \{B\}$, and B is the key of N_2

  - Conclusion (a wrong one): The set of relation schemas D is a lossless join decomposition (of a universal relation schema)
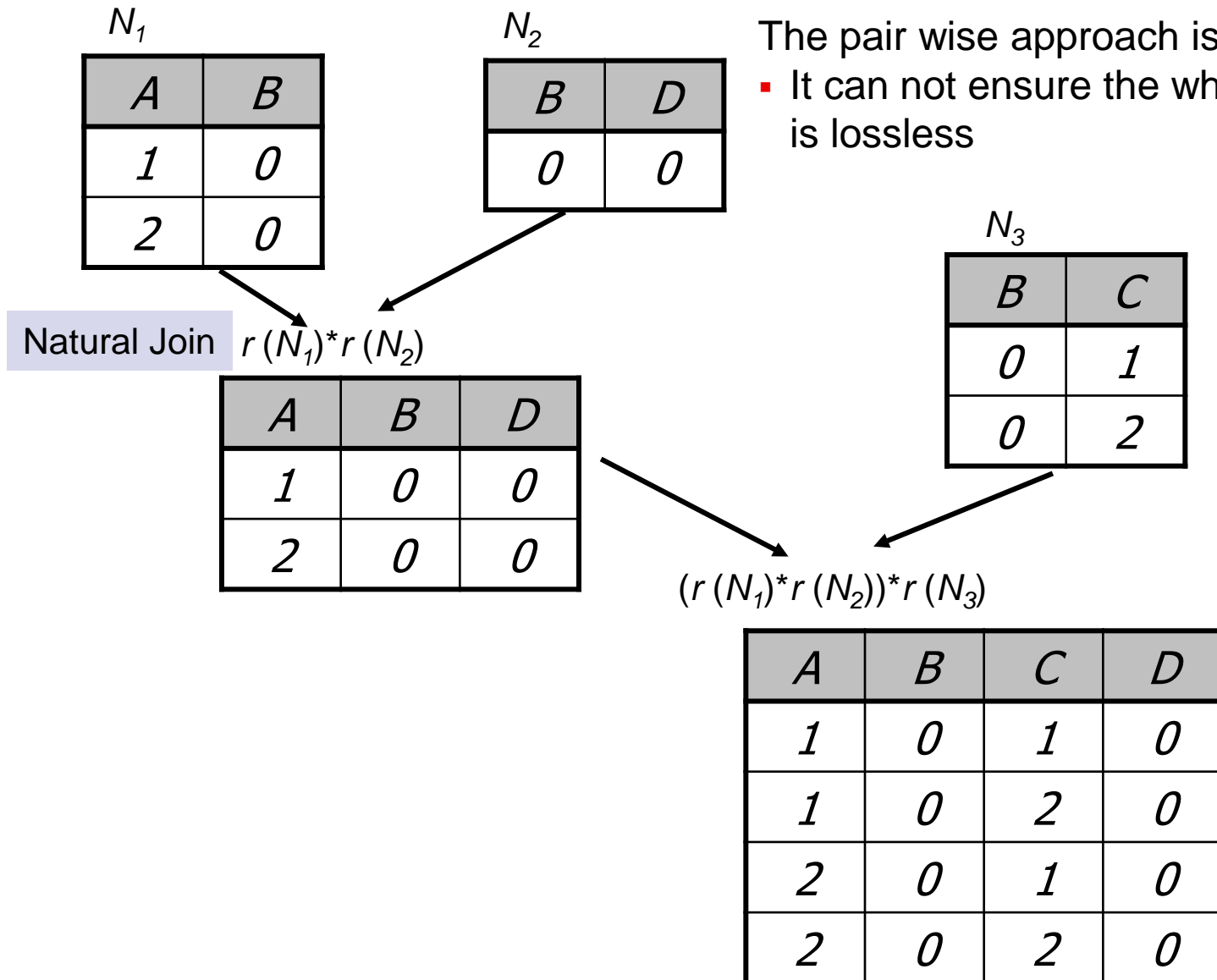
$D = \{N_1(\{A, B\}, \{A\}), N_2(\{B, D\}, \{B\}), N_3(\{C, B\}, \{C\})\}$

- A correct approach is to apply this checking iteratively until all the schemas are considered
  - $\{A, B\} \cap \{B, D\} = \{B\}$, and B is the key of N_2,
  - construct new relation schema $N_{12}(R_{12}, \text{Key}(N_{12}))$, with $R_{12} = \{A, B\} \cup \{B, D\} = \{A, B, D\}$ and $\text{Key}(N_{12}) = \{A\}$

  - $\{A, B, D\} \cap \{B, C\} = \{B\}$, and check again.
  - B is neither a key of N_{32} nor a key of N_3

- We can conclude the set of relation schemas D is a not a lossless join decomposition (of a universal relation schema).

Universal Relation Schema

A	B	C	D
1	0	1	0
2	0	2	0

p r o j e c t i o n

π_{AB} π_{BD} π_{BC}

N_1

$\underline{A}$	B
1	0
2	0

N_2

$\underline{B}$	D
0	0

N_3

B	$\underline{C}$
0	1
0	2

N_1

A	B
1	0
2	0

N_2

B	D
0	0

The pair wise approach is wrong since
- It can not ensure the whole decomposition is lossless

N_3

B	C
0	1
0	2

Natural Join $r(N_1)*r(N_2)$

A	B	D
1	0	0
2	0	0

$(r(N_1)*r(N_2))*r(N_3)$

A	B	C	D
1	0	1	0
1	0	2	0
2	0	1	0
2	0	2	0

One Approach of Checking Losslessness of D

- To check whether a set D of relation schemas is a lossless decomposition is:

    1. Construct a relation schema (U, F), where

    $$U = \bigcup_{i=1}^{n} R_i \quad \text{and} \quad F = \bigcup_{i=1}^{n} F_i$$

    2. Find all keys $\{X_i \mid i = 1,..., m\}$ of the constructed "universal" relation schema (U, F)

    3. If there is a relation schema $N(R, K)$ in D that contains a key of the constructed relation schema (U, F), then D is a lossless join decomposition

    4. Otherwise, add a new relation schema that contains only a key X_i of the constructed "universal" relation schema (U, F) to D

    $$D = D \cup \{N_x(X_i, X_i)\}$$

Example 2: Checking Losslessness of D

- The universal relation schema key is *AC*, and decompositions

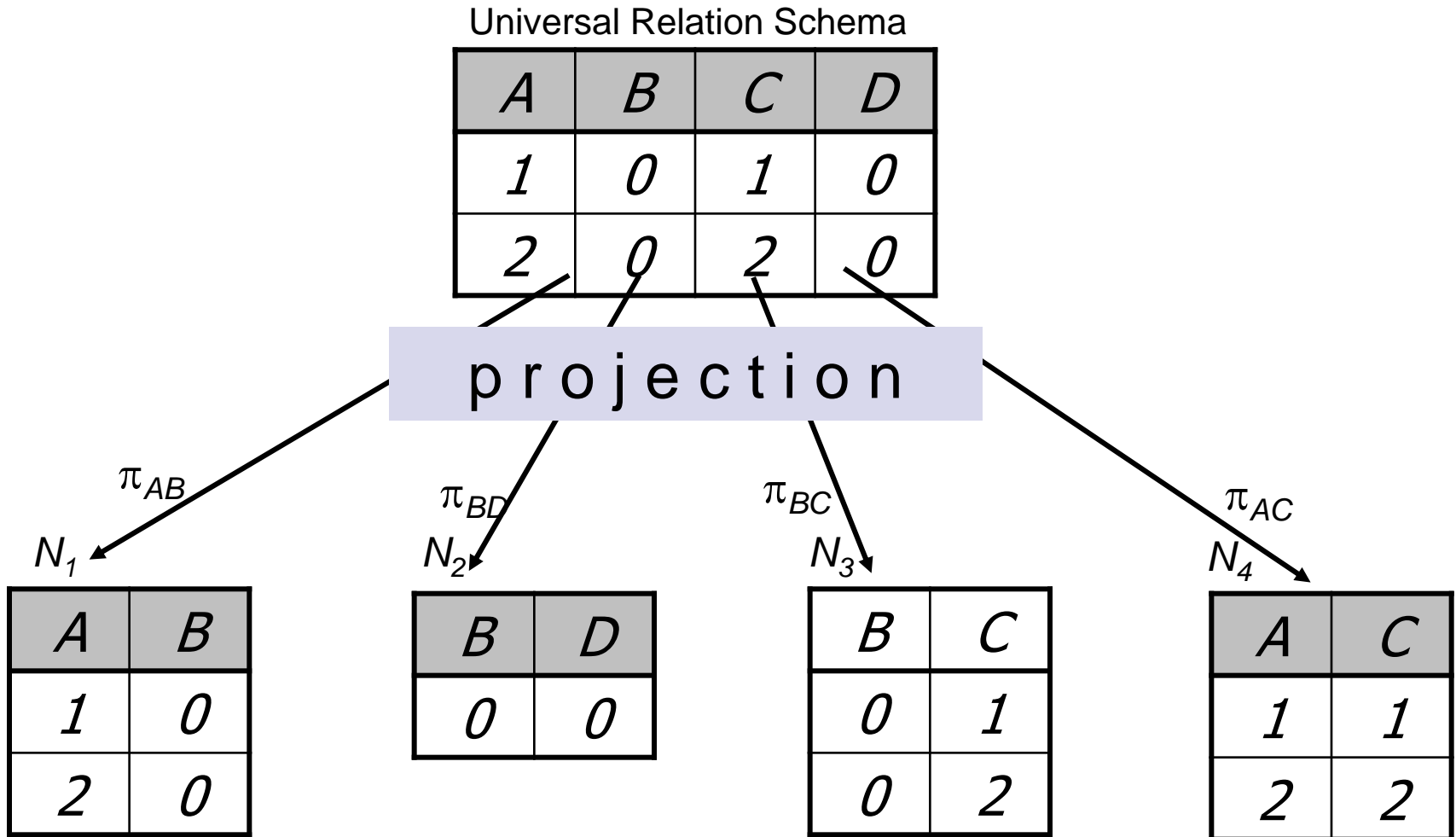 $D = \{N_1(\{A, B\}, \{A\}), N_2(\{B, D\}, \{B\}), N_3(\{B, C\}, \{C\}))\}$

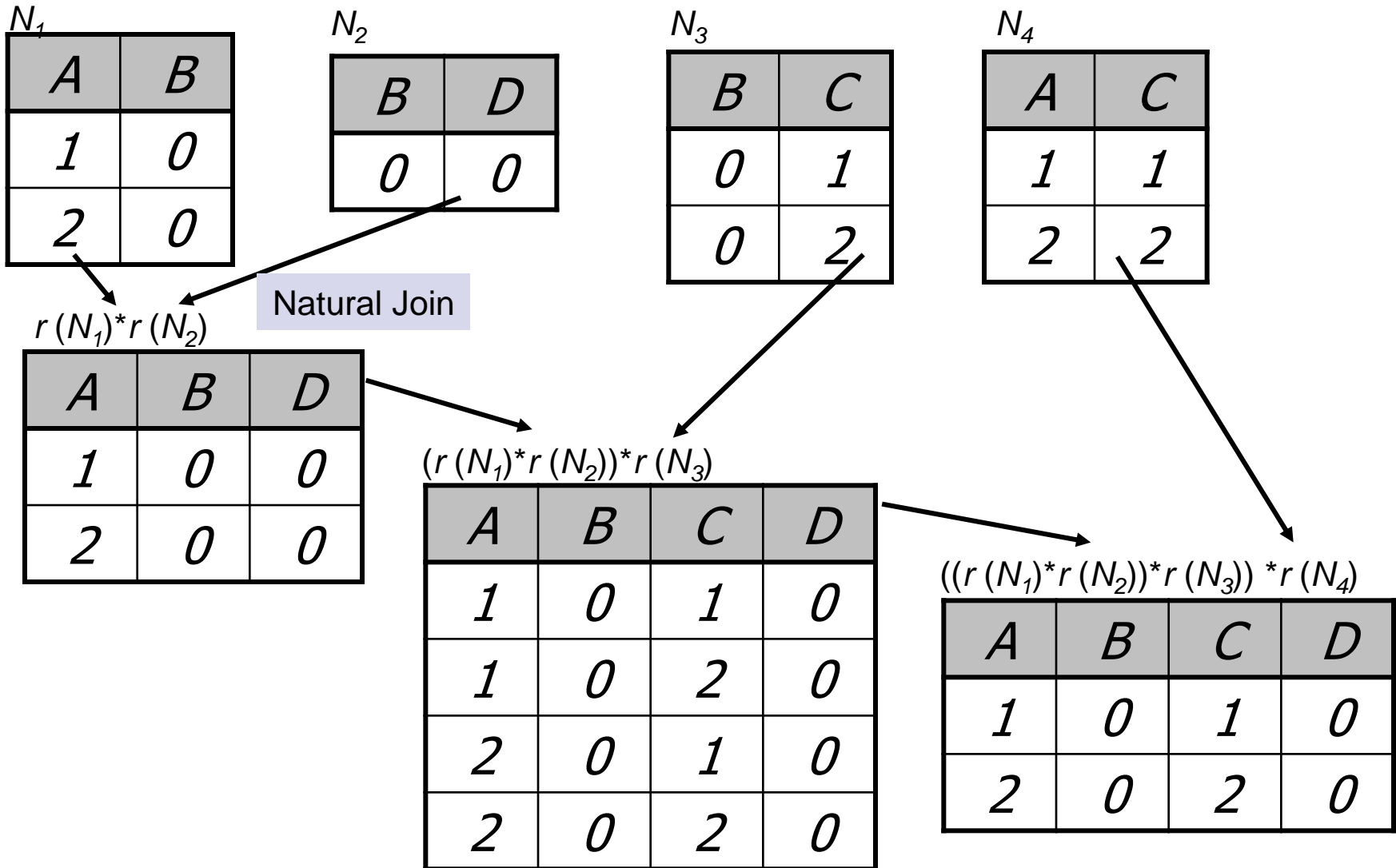 Is the decomposition lossless? Why?

- A lossless decomposition

  $D' = \{N_1(\{A, B\}, \{A\}), N_2(\{B, D\}, \{B\}), N_3(\{B, C\}, \{C\}),$
  $\qquad N_4(\{A, C\}, \{AC\})\}$

- This can be achieved using the Synthesis Algorithm

Universal Relation Schema

A	B	C	D
1	0	1	0
2	0	2	0

p r o j e c t i o n

π_{AB}

π_{BD}

π_{BC}

π_{AC}

N_1

A	B
1	0
2	0

N_2

B	D
0	0

N_3

B	C
0	1
0	2

N_4

A	C
1	1
2	2

N_1

A	B
1	0
2	0

N_2

B	D
0	0

N_3

B	C
0	1
0	2

N_4

A	C
1	1
2	2

Natural Join

$r(N_1)*r(N_2)$

A	B	D
1	0	0
2	0	0

$(r(N_1)*r(N_2))*r(N_3)$

A	B	C	D
1	0	1	0
1	0	2	0
2	0	1	0
2	0	2	0

$((r(N_1)*r(N_2))*r(N_3))*r(N_4)$

A	B	C	D
1	0	1	0
2	0	2	0

- Consider the following relation schema again

  Department ({*LecId*, *LeName*, *CourId*, *CoName*, *DptId* }, {*LecId* →*LeName* + *CourId*, *CourId* →*CoName* + *DptId* })

- Is Department in 3NF? If not, decompose it into 3NF

- Is Department in BCNF? If not, decompose it into BCNF

Exercise 1: 3NF Decomposition

- Consider the following relation schema again

  Department ({*LecId, LeName, CourId, CoName, DptId* },
  {*LecId* →*LeName* + *CourId, CourId* →*CoName* + *DptId* })

- Compute minimal cover of *F*

  G= {*LecId* →*LeName* , *LecId* → *CourId, CourId* →*CoName,*

  CourId → *DptId* })

1. Group FDs according to LHS and form relation schemas

   LecId →*LeName* , *LecId* → *CourId,*

   CourId →*CoName,* *CourId* → *DptId*

   Lecturer ({*LecId, LeName, CourId* }, {*LecId* })
   Course ({*CourId, CoName, DptId* }, {*CourId* })

Exercise 1: 3NF Decomposition

2. Compute universal relation keys and check if any of the relation schemas contains one of the keys:

   ▪ *LecId* is the universal relation schema key and is in Lecturer

3. All functional dependencies are preserved

Exercise 1: BCNF Decomposition

- Consider the following relation schema again

  Department ({*LecId, LeName, CourId, CoName, DptId* },
  {*LecId* → *LeName* + *CourId, CourId* → *CoName* + *DptId* })

  Is *Department* in BCNF?

- Is decomposed into BCNF using *CourId* → *CoName* + *DptId*

  Lecturer ({*LecId, LeName, CourId* }, {*LecId* })

  Course ({*CourId, CoName, DptId* }, {*CourId* })

- Both Lecturer and Course are in BCNF

- Let *R = ABCD* a relation schema and

  F ={AB → C, C → D, D → A} a set of dependencies for *R*.

1) Find the candidate keys for *R*

2) If *R* is not in BCNF, give a decomposition of *R* in relations that will be in BCNF

Exercise 3: Find Keys and Normalization

- Let $R = JKL$ a relation and $F = \{JK \rightarrow L, L \rightarrow K\}$ a set of dependencies for R.

- Find two canditate keys in R

- Is R in 3NF? Justify your answer

- If R is not in BCNF, decompose R into BCNF

- Are the functional dependencies preserved during the decomposition?