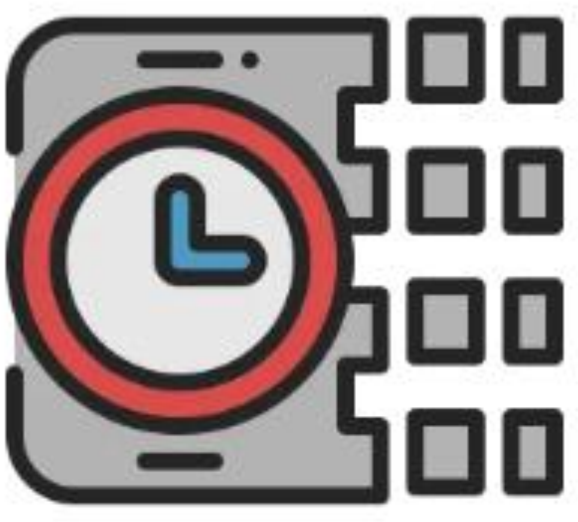
Desirable Properties of Decomposition



Module No. 3

Functional dependency (FD), Closure of FD, Closure of Attributes, Cover, Equivalence of FD, Canonical cover, Key generation, Normalization, Desirable properties of decomposition.

Decomposition of a Relation

Properties of Decomposition

The following two properties must be followed when decomposing a given relation

1. Lossless Decomposition

Lossless decomposition ensures-

1. No information is lost from the original relation during decomposition.
2. When the sub-relations are joined back, the same relation is obtained that was decomposed. Every decomposition must always be lossless.

Decomposition of a Relation

Properties of Decomposition

2. Dependency Preservation

Dependency preservation ensures:

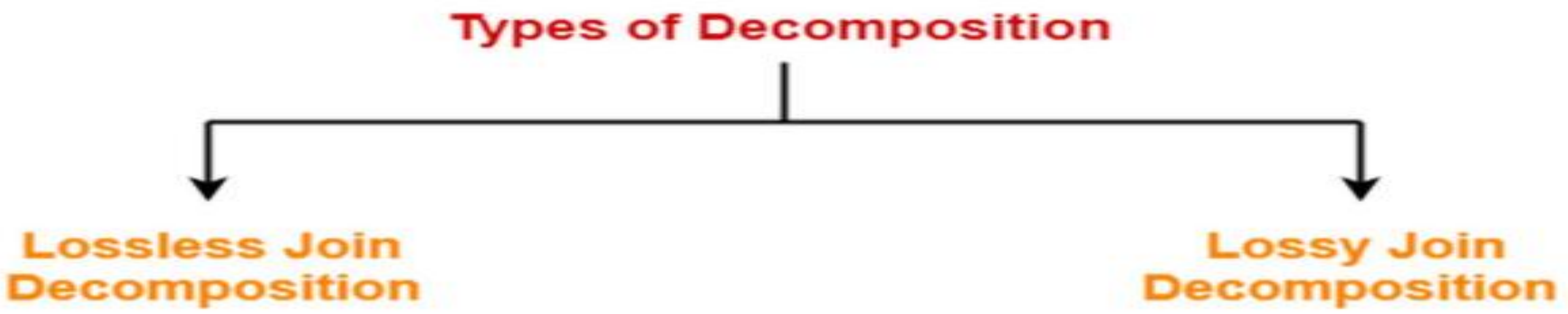
- None of the functional dependencies that hold on the original relation are lost.
- The sub-relations still hold or satisfy the functional dependencies of the original relation.

- Let a relation R (A,B,C,D) and a set of FDs F = { A->B,A->C,C->D} are given.
- A relation is decomposed into –
- R1 = (A, B, C) with FDs F1 = {A->B, A->C}.
- R2 = (C, D) with FDs F2 = {C->D}
- F' = F1 U F2 = {A->B, A->C, C->D} So, F'= F. And so, F'+ = F+.
- Thus, the decomposition is dependency-preserving decomposition.

Decomposition of a Relation

- The process of breaking up or dividing a single relation into two or more sub-relations is called as decomposition of a relation.

Types of Decomposition



Decomposition of a Relation

1.Lossless Join Decomposition

- Consider there is a relation R which is decomposed into subrelations R1 , R2 , . . ., Rn.
- This decomposition is called lossless join decomposition when the join of the sub-relations results in the same relation R that was decomposed.
- For lossless join decomposition, we always have- .

$$\boxed{R_1 \bowtie R_2 \bowtie R_3 \ldots\ldots \bowtie R_n = R}$$

where $\bowtie$ is a natural join operator

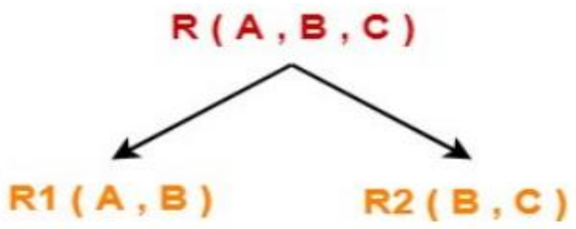
Decomposition of a Relation

1.Lossless Join Decomposition

Example: Consider the following relation R(A , B , C)

A	B	C
1	2	1
2	5	3
3	3	3

Consider this relation is decomposed into two sub relations R1(A , B) and R2(B , C)



The two sub relations are-

A	B
1	2
2	5
3	3

R_1(A , B)

B	C
2	1
5	3
3	3

R_2(B , C)

Decomposition of a Relation

1.Lossless Join Decomposition

Example:

R (A , B , C)

R1 (A , B)   R2 (B , C)

The two sub relations are-

A	B
1	2
2	5
3	3

R_1(A , B)

B	C
2	1
5	3
3	3

R_2(B , C)

- Now, let us check whether this decomposition is lossless or not. For lossless decomposition, we must have R1 ⋈ R2 = R.
- Now, if we perform the natural join (⋈) of the sub relations R1 and R2 , we get

A	B	C
1	2	1
2	5	3
3	3	3

Decomposition of a Relation

1.Lossless Join Decomposition

Lossless join decomposition is also known as non-additive join decomposition.

- This is because the resultant relation after joining the sub-relations is the same as the decomposed relation.
- No extraneous tuples appear after joining of the sub-relations.

Decomposition of a Relation

2. Lossy Join Decomposition

- Consider there is a relation R which is decomposed into sub-relations R1 , R2 , …. , Rn.
- This decomposition is called lossy join decomposition when the join of the sub-relations does not result in the same relation R that was decomposed.
- The natural join of the sub-relations is always found to have some extraneous tuples.

$$R_1 \bowtie R_2 \bowtie R_3 \ldots\ldots \bowtie R_n \supset R$$

where $\bowtie$ is a natural join operator

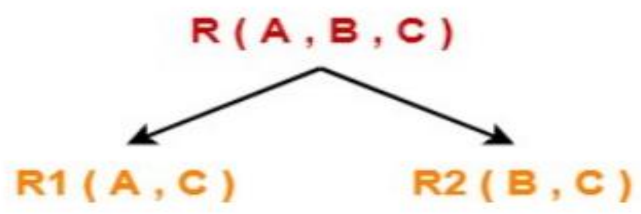
Decomposition of a Relation

2. Lossy Join Decomposition

Example   Consider the following relation R(A , B , C)

A	B	C
1	2	1
2	5	3
3	3	3

Consider this relation is decomposed into two sub relations R1(A , C) and R2(B , C)



The two sub relations are-

A	C
1	1
2	3
3	3

R_1(A , B)

B	C
2	1
5	3
3	3

R_2(B , C)

Decomposition of a Relation

2. Lossy Join Decomposition

The two sub relations are-

A	C
1	1
2	3
3	3

R_1(A , B)

B	C
2	1
5	3
3	3

R_2(B , C)

Example

- Now, let us check whether this decomposition is lossy or not.
- For lossy decomposition, we must have **R1 ⋈ R2 ⊃ R**

Now, if we perform the natural join (⋈) of the sub relations R1 and R2

A	B	C
1	2	1
2	5	3
2	3	3
3	5	3
3	3	3

- This relation is not the same as the original relation R and contains some extraneous tuples.
- Clearly, R1 ⋈ R2 ⊃ R. Thus, we conclude that the above decomposition is lossy join decomposition.

Decomposition of a Relation

2. Lossy Join Decomposition

NOTE-

- Lossy join decomposition is also known as careless decomposition.
- This is because extraneous tuples get introduced in the natural join of the sub-relations.
- Extraneous tuples make the identification of the original tuples difficult.

Decomposition of a Relation

Determining Whether Decomposition is Lossless or Lossy

Consider a relation R is decomposed into two sub relations R1 and R2. Then,

- If all the following conditions are satisfied, then the decomposition is lossless.
- If any of these conditions fail, then the decomposition is lossy.

Condition-01

- The Union of both the sub-relations must contain all the attributes that are present in the original relation R. Thus

$R1 \cup R2 = R$

Condition-02

- The intersection of both the sub relations must not be null. In other words, there must be some common attribute that is present in both the sub relations. Thus

$R1 \cap R2 \neq \emptyset$

Condition-03

- Intersection of both the sub relations must be a super key of either R1 or R2 or both. Thus

$R1 \cap R2$ = Super key of R1 or R2

Decomposition of a Relation

Examples

- Consider a relation schema R (A , B , C , D) with the functional dependencies $A \rightarrow B$ and $C \rightarrow D$. Determine whether the decomposition of R into R1 (A , B) and R2 (C , D) is lossless or lossy.

Solution

Condition-01: The union of both the sub relations must contain all the attributes of relation R.

R1 (A , B) ∪ R2 (C , D) = R (A , B , C , D)

- The union of the sub-relations contains all the attributes of relation R. Thus, condition-01 satisfies.

Condition-01: the intersection of both the sub relations must not be null.

R1 (A , B) ∩ R2 (C , D) = Φ

- The intersection of the sub-relations is null.
- So, condition-02 fails.
- Thus, we conclude that the decomposition is lossy.

Decomposition of a Relation

Examples

- Consider a relation schema R (A , B , C , D) with the following functional dependencies.

FD : $A \rightarrow B$, $B \rightarrow C$, $C \rightarrow D$, $D \rightarrow B$

- Determine whether the decomposition of R into R1 (A , B) , R2 (B , C) and R3 (B , D) is lossless or lossy.

Strategy to Solve

When a given relation is decomposed into more than two sub relations, then-

- Consider any one possible way in which the relation might have been decomposed into those sub-relations.
- First, divide the given relation into two sub-relations.
- Then, divide the sub-relations according to the sub-relations given in the question.

As a thumb rule,

remember Any relation can be decomposed only into two sub-relations at a time.

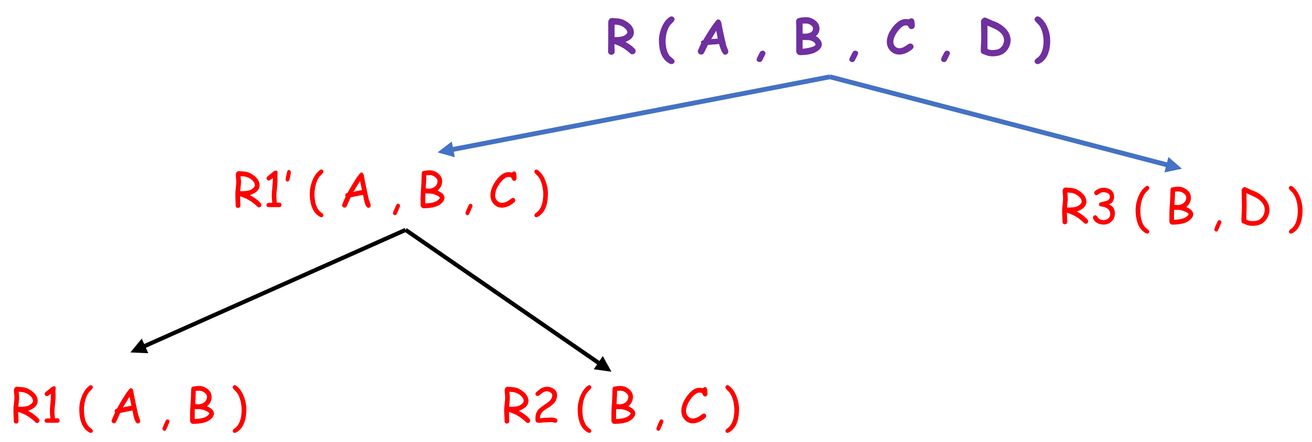
Decomposition of a Relation

Examples

- Consider a relation schema R (A , B , C , D) with the following functional dependencies.

FD : $A \rightarrow B$, $B \rightarrow C$, $C \rightarrow D$, $D \rightarrow B$

- Determine whether the decomposition of R into R1 (A , B) , R2 (B , C) and R3 (B , D) is lossless or lossy.



Condition-01

The union of both the sub-relations must contain all the attributes of relation R.

R' (A , B , C) ∪ R3 (B , D) = R (A , B , C , D)

- Condition-01 satisfies.

Decomposition of a Relation

Examples Cont'd

- $R (A , B , C , D)$

FD : $A \rightarrow B$, $B \rightarrow C$, $C \rightarrow D$, $D \rightarrow B$

- Determine whether the decomposition of R into R1 (A , B) , R2 (B , C) and R3 (B , D) is lossless or lossy.

Condition-02: The intersection of both the sub relations must not be null.

$$R' (A , B , C) \cap R3 (B , D) = B$$

Condition-02 satisfies

Condition-03:

- The intersection of both the sub-relations must be the super key of one of the two sub-relations or both.

$$R' (A , B , C) \cap R3 (B , D) = B$$

the closure of attribute B is $B+ = \{ B , C , D \}$

- Attribute 'B' can not determine attribute 'A' of sub relation R'.
- Thus, it is not a super key of the sub relation R'.
- Attribute 'B' can determine all the attributes of sub relation R3.
- Thus, it is a super key of the sub relation R3

condition-03 satisfies

Decomposition of a Relation

Examples Cont'd

- $R (A , B , C , D)$

FD : $A \to B$, $B \to C$, $C \to D$, $D \to B$

- Determine whether the decomposition of R into R1 (A , B) , R2 (B , C) and R3 (B , D) is lossless or lossy.

Decomposition of R'(A, B, C) into R1(A, B) and R2(B, C)-

Condition-01: According to condition-01, the union of both the sub relations must contain all the attributes of relation R′. So,

$$R1 (A , B) \cup R2 (B , C) = R' (A , B , C)$$

Clearly, the union of the sub relations contain all the attributes of relation R′.

Thus, condition-01 satisfies.

Condition-02: According to condition-02, intersection of both the sub relations must not be null. So, we have-

$$R1 (A , B) \cap R2 (B , C) = B$$

Clearly,

intersection of the sub-relations is not null. Thus, condition-02 satisfies.

Decomposition of a Relation

Examples Cont'd

- R (A , B , C , D)

FD : $A \rightarrow B$, $B \rightarrow C$, $C \rightarrow D$, $D \rightarrow B$

- Determine whether the decomposition of R into R1 (A , B) , R2 (B , C) and R3 (B , D) is lossless or lossy.

Decomposition of R'(A, B, C) into R1(A, B) and R2(B, C)-

Condition-03:

According to condition-03, intersection of both the sub relations must be the super key of one of the two sub-relations or both. So, we have-

$$R1 (A , B) \cap R2 (B , C) = B$$

Now, the closure of attribute B is $B^+ = \{ B , C , D \}$

- Attribute 'B' can not determine attribute 'A' of sub relation R1.
- Thus, it is not a super key of the sub relation R1.
- Attribute 'B' can determine all the attributes of sub relation R2.
- Thus, it is a super key of the sub relation R2.

Condition-03 satisfies.
The decomposition is lossless.

Decomposition of a Relation

Try this

EmployeeProjectDetail

Employee_Code	Employee_Name	Employee_Email	Project_Name	Project_ID
101	John	john@demo.com	Project103	P03
101	John	john@demo.com	Project101	P01
102	Ryan	ryan@example.com	Project102	P02
103	Stephanie	stephanie@abc.com	Project102	P02

EmployeeProject

Employee _Code	Project_ID	Employee _Name	Employee_Email
101	P03	John	john@demo.com
101	P01	John	john@demo.com
102	P04	Ryan	ryan@example.com
103	P02	Stephanie	stephanie@abc.com

The primary key of the above relation is {Employee_Code, Project_ID}.

ProjectDetail

Project_ID	Project_Name
P03	Project103
P01	Project101
P04	Project104
P02	Project102

The primary key of the above relation is {Project_ID}.

Decomposition of a Relation

Dependency Preservation

Example:

$R=(A, B, C)$, $F=\{A \rightarrow B, B \rightarrow C\}$

Decomposition of R: $R1=(A, C)$ $R2=(B, C)$

Does this decomposition preserve the given dependencies?

Solution:

In R1 the following dependencies hold: $F1'=\{A \rightarrow A, C \rightarrow C, A \rightarrow C, AC \rightarrow AC\}$

In R2 the following dependencies hold: $F2'= \{B \rightarrow B, C \rightarrow C, B \rightarrow C, BC \rightarrow BC\}$

The set of nontrivial dependencies hold on R1 and R2: $F' := \{B \rightarrow C, A \rightarrow C\}$

$A \rightarrow B$ can not be derived from F', so this decomposition is NOT dependency preserving.

Decomposition of a Relation

Dependency Preservation

Dependency preservation

Example:

R=(A, B, C), F={A→B, B→C}

Decomposition of R: R1=(A, B) R2=(B, C)

Does this decomposition preserve the given dependencies?

Solution:

In R1 the following dependencies hold: F1={A→B, A→A, B→B, AB→AB}

In R1 the following dependencies hold: F2= {B→B, C→C, B→C, BC→BC}

F'= F1' ∪ F2' = {A→B, B→C, trivial dependencies}

In F' all the original dependencies occur, so this decomposition preserves dependencies.

Decomposition of a Relation

Dependency Preservation

Example:

R(A, B, C, D), $F=\{A \rightarrow B,\ B \rightarrow C\}$
Let S(A,C) be a decomposed relation of R. What dependencies do hold on S?

Solution: Need to compute the closure of each subset of $\{A,C\}$, wrt F^+

Compute $\{A\}+ = \{ABC\}$
- C is in S
- so $A \rightarrow C$ holds for S

Compute $\{C\}+$
- $\{C\}+ = C$, no new FD

Compute $\{AC\}+$
- $\{AC\}+ = ABC$, no new FD

Hence, $A \rightarrow C$ is the only non-trivial FD for S, $\Pi_S(F^+)=\{A\rightarrow C, + \text{trivial FDs}\}$

Decomposition of a Relation

Dependency Preservation

Example:

R(A, B, C, D), F={A $\rightarrow$ B, B $\rightarrow$ C}
Let S(A,C) be a decomposed relation of R. What dependencies do hold on S?

Solution: Need to compute the closure of each subset of {A,C}, wrt F^+

Compute {A}+ = {ABC}
- C is in S
- so A $\rightarrow$ C holds for S

Compute {C}+
- {C}+ = C, no new FD

Compute {AC}+
- {AC}+ = ABC, no new FD

Hence, A $\rightarrow$ C is the only non-trivial FD for S, $\Pi_S(F^+)$={A$\rightarrow$C, + trivial FDs}

Decomposition of a Relation

Dependency Preservation

Example:

R(A, B, C, D, E), $A \rightarrow D$, $B \rightarrow E$, $DE \rightarrow C$.

Let S(A, B, C) be a decomposed relation of R. What FD-s do hold on S?

Solution: Need to compute the closure of each subset of {A, B, C}

Compute $\{A\}+ = AD$, $A \rightarrow D$, no new FD

Compute $\{B\}+ = BE$, but E is not in S, so $B \rightarrow E$ does not hold

Compute $\{C\}+ = C$, no new FD

Compute $\{AB\}+ = ABCDE$, so $AB \rightarrow C$ holds for S (since DE are not in S)

Compute $\{BC\}+ = BCE$, no new FD

Compute $\{AC\}+ = ACD$, no new FD

Compute $\{ABC\}+ = ABCDE$, no new FD

Hence, $AB \rightarrow C$ is the only nontrivial FD for S, so $\Pi_S(F^+)=\{A \rightarrow C, +$ trivial FDs$\}$

Dependency Preservation

Try

1. R (A, B, C, D) is decomposed into R1(A, B, C), R2(C, D) and $F=\{B \rightarrow C, AC \rightarrow D\}$.

What dependencies do hold in R1 and in R2?

Hint: Find the following closures:

$\{A\}^+=$

$\{B\}^+=$

$\{C\}^+=$

$\{A,B\}^+=$

$\{A,C\}^+=$

$\{A,D\}^+=$

$\{B,C\}^+=$

$\{B,D\}^+=$

$\{C,D\}^+=$

$\{A,B,C\}^+=$

$\{A,B,D\}^+=$

$\{B,C,D\}^+=$

$\{A,C,D\}^+=$