Decomposition of Functional Relations
Examples

Closure Algorithm

Example 1. Consider a relation $R(A, B, C, D, E, F)$ with asserted FDs

(1)	$A \rightarrow B, C$
(2)	$F \rightarrow E$
(3)	$B, F, E \rightarrow D$
(4)	$A, D \rightarrow E$
(5)	$C, E, \rightarrow D, F$

Problem 1: Find $\{A\}^+$.

Solution. We start with set $X_0 = \{A\}$.

The left side of FD (1) is $A \subseteq X_0$. We set $X_1 = \{A\} \cup \{B, C\} = \{A, B, C\}$.

The left sides of FDs (2),(3),(4),(5) are not proper subsets of X_1. Therefore, $\{A\}^+ = X_1 = \{A, B, C\}$.

Problem 2: Find $\{A, D\}^+$.

Solution. We start with set $X_0 = \{A, D\}$.

The left side of FD (1) is $\{A\} \subseteq X_0$. We set $X_1 = \{A\} \cup \{B, C\} = \{A, B, C\}$.

The left side of FD (4) is $\{A, D\} \subseteq X_1$. We set $X_2 = X_1 \cup \{E\} = \{A, B, C, D, E\}$.

The left side of FD (5) is $\{C, E\} \subseteq X_2$. We set $X_3 = X_2 \cup \{D, F\} = \{A, B, C, D, E, F\}$.

X_3 contains all attributes from R, therefore $\{A, D\}^+ = X_3 = \{A, B, C, D, E, F\}$.

FD Projection Algorithm

Example 2. Consider a relation $R(A, B, C, D, E, F)$ with asserted FDs

(1)	$A \rightarrow B, C$
(2)	$F \rightarrow E$
(3)	$B, F, E \rightarrow D$
(4)	$A, D \rightarrow E$
(5)	$C, E, \rightarrow D, F$

Problem 1: Find FDs asserted on $R_1 = \pi_{A,B,C,D}(R)$

Solution. Start with $S_0 = \emptyset$. We consider all subsets of $\{A, B, C, D\}$ in turn.

$\{A\}$: left side of FD (1) is A. Right side of FD (1) is $\{B, C\} \subseteq \{A, B, C, D\}$. $S_1 = S_0 \cup \{A \to B, C\}$. No other FD can be matched.

$\{B\}$. No FD has left side B.

$\{C\}$. No FD has left side C.

$\{D\}$. No FD has left side D.

$\{A, B\}$. FD (1) qualifies, but we have already dealt with it. No new FDs emerge.

$\{A, C\}$. No new FDs emerge.

$\{A, D\}$. FD (4) has left side A, D. It's right side is $\{E\} \cap \{A, B, C, D\} = \emptyset$, hence no new FDs are added to S_1.

$\{B, C\}$, $\{B, D\}$, $\{C, D\}$: no FDs emerge.

$\{A, B, C\}, \{B, C, D\}, \{A, C, D\}, \{A, B, D\}$: no new FDs emerge.

Therefore, there is only one FD asserted on R_1: $A \to B, C$.

BCNF Decomposition

Example 3. Consider a relation $R(A, B, C, D, E, F)$ with asserted FDs

(1)	$A \to C, D$
(2)	$B \to E$
(3)	$A, E \to F$

Let us decompose R into BCNF.

Step 1. Establish all keys of R. Using FD-Closure algorithm, we can establish that A, B is the only key of R.

Step 2. Check if R is in BCNF. We observe that none of the FDs above satisfy BCNF condition, as neither A, nor B, nor A, E are superkeys.

Step 3. Decompose R. We pick FD (1). $\{A\}^+ = \{A, C, D\}$. We decompose R into $R1(A, C, D)$, and $R2(A, B, E, F)$.

Step 3. Decompose $R1$. Using FD-Project we assert $A \to C, D$ FD on $R1$, and verify that A is a key. $R1$ is in BCNF.

Step 4. Decompose $R2$. Using FD-Project we assert $B \to E$ and $A, E \to F$ on $R2$.

Using FD-Closure we establish that A, B is the only key. Then FD (2) violates BCNF.

$\{B\}^+ = \{B, E\}$. We decompose $R2$ into $R3(B, E)$ and $R4(A, B, F)$.

Step 5. Decompose $R3$. Using FD-Project we assert $B \to E$ on $R3$. B is the key, and $R3$ is in BCNF.

Step 6. Decompose $R4$. Using FD-Project we assert $A, B \to F$ on $R4$. A, B is the key, and R is in BCNF.

Therefore, the decomposition of $R(A, B, C, D, E, F)$ into a BCNF schema is: $R1(A, C, D)$, $R3(B, E)$, $R4(A, B, E)$.

Example 4. BCNF decomposition gone awry. Consider a relation $R(A, B, C)$ with the following FDs asserted:

(1)	$A \to B$
(2)	$B, C \to A$

Task: Decompose R into BCNF.

R is NOT in BCNF. R has two keys: B, C and A, C. FD (1) violates BCNF condition (A is not a superkey). BCNF decomposition yields two relations:

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

Both relations are in BCNF.

Problem: We assert $A \rightarrow B$ on $R1$. However, FD (2) cannot be asserted on either of the tables. This may lead to the following problem.

Consider the following instances of $R1$ and $R2$:

R1:

A	B
a	b
b	b

R2:

B	C
b	a
b	d

Let us compute $R1 \bowtie R2$:

$R1 \bowtie R2$:

A	B	C
a	b	a
a	b	d
b	b	a
b	b	d

We note the the FD $A \rightarrow B$ holds on $R1 \bowtie R2$. However, the FD $B, C \rightarrow A$ **does not**. Indeed, tuples (a,b,a) and (b,b,a) agree on values of B, C but NOT on values of A.

We conclude that the BCNF decomposition of R **does not preserve all functional dependencies**.

Example 5. BCNF decomposition gone awry part 2. Consider the relation Schedule(Classroom, Day, Course, Time, Instructor) with with the following FDs asserted:

(1) Classroom, Day, Time→ Instructor, Course
(2) Instructor, Course → Classroom, Time

In previous examples we learned that this relation has two keys: Classroom, Day, Time, and Instructor, Course, Day.

Because every attribute of this table is *prime*, Schedule is in 3NF.

Because the left side of the FD (2) does not contain a key, Schedule is NOT in BCNF.

A BCNF decomposition would yield two tables:

Schedule1(Instructor, Course, Classroom, Time)
Schedule2(Instructor, Course, Day)

FD (2) is preserved on table Schedule1.

Because the left side of FD (1) is not found in any table, FD (1) "disappears." Consider the following instances of the tables Schedule1 and Schedule2.

Schedule1:

Instructor	Course	Classroom	Time
Dekhtyar	CSC 366	14-253	12:10
Dekhtyar	CSC 468	14-252	12:10

Schedule2:

Instructor	Course	Day
Dekhtyar	CSC 366	T
Dekhtyar	CSC 366	R
Dekhtyar	CSC 468	T
Dekhtyar	CSC 468	R

This combination of relational tables is consistent, as no FDs are violated. However, Schedule1⋈ Schedule2 yields:

Schedule1:

Instructor	Course	Classroom	Time	Day
Dekhtyar	CSC 366	14-253	12:10	T
Dekhtyar	CSC 366	14-253	12:10	R
Dekhtyar	CSC 468	14-252	12:10	T
Dekhtyar	CSC 468	14-252	12:10	R

This table violates FD (1) from above.

3NF Decomposition

Example 6. 3NF Decomposition. Consider a relational table $R(A, B, C, D, E, F)$ with the following FDs asserted:

$A \rightarrow B$
$A \rightarrow C$
$C \rightarrow D$
$D \rightarrow F$
$E, B \rightarrow F$

First, let us note that the set of FDs above forms a minimal basis. (try removing any FD, and deriving it from the rest).

Second, we note that R is not in 3NF, due to the presence of transitive FDs.

We use 3NF-Decompose algorithm to create the following tables:

R1(A,B)
R2(A,C)
R3(C,D)
R4(D,F)
R5(B,E,F)
R6(A,E)

Note, that each table $R1 - R6$ is in 3NF, and each FD from above is preserved in at least one table.

Note, also, that this is not the most concise decomposition of R. In fact, we can combine $R1$ and $R2$ into a single relation $R12(A, B, C)$.

Finally, we note that $R6$ has been created because none of $R1 - R5$ contained a full key of R.