Normalization

Functional Dependency and Schema Refinement

Canonical cover

What is canonical cover?

- A canonical cover of F is a **minimal set of functional dependencies** equivalent to F, having **no redundant dependencies or redundant parts of dependencies**.
- It is denoted by **F_c**
- A canonical cover for F is a set of dependencies F_c such that
  - **F logically implies** all dependencies in F_c and
  - **F_c logically implies** all dependencies in F and
  - **No** functional dependency in F_c contains an **extraneous attribute** and
  - Each **left side** of functional dependency in F_c is **unique**.

Decomposition Rule

$F = \{A \rightarrow B, A \rightarrow C\}$

$F_c = \{A \rightarrow BC\}$

Union Rule

Algorithm to find canonical cover

- *Repeat*
  - Use the **union rule** to replace any dependencies in F **$\alpha 1 \rightarrow \beta 1$** and **$\alpha 1 \rightarrow \beta 2$ with $\alpha 1 \rightarrow \beta 1 \beta 2$**
  - Find a functional dependency $\alpha \rightarrow \beta$ with an **extraneous attribute** either in α or in β

    /* Note: test for extraneous attributes done using F_c, not F */
    - If an **extraneous attribute is found, delete it** from $\alpha \rightarrow \beta$
- *until* F does not change

  /* Note: Union rule may become applicable after some extraneous attributes have been deleted, so it has to be re-applied */

Canonical cover [Example]

- Consider the relation schema R = (A, B, C) with FDs

  F = {A → BC, B → C, A → B, AB → C}

- Find canonical cover.

- Combine A → BC and A → B into A → BC (Union Rule)
  - Set is {A → BC, B → C, AB → C}
- A is extraneous in AB → C
  - Check if the result of deleting A from AB → C is implied by the other dependencies
    - Yes: in fact, B → C is already present
  - Set is {A → BC, B → C}
- C is extraneous in A → BC
  - Check if A → C is logically implied by A → B and the other dependencies
    - Yes: using transitivity on A → B and B → C.
  - The canonical cover is: **A → B, B → C**

Canonical cover [Example]

- Consider the relation schema R = (A, B, C, D, E, F) with FDs

  F = {B → A, AD → BC, C → ABD}

- Find canonical cover.

- Decompose RHS: B → A, AD → B, AD → C, C → A, C → B, C → D
- Find the closure of each FD by including and removing that FD
  - Considering B → A
    - Including $B^+ = \{BA\}$ B → A, AD → B, AD → C, C → A, C → B, C → D
    - Excluding $B^+ = \{B\}$ AD → B, AD → C, C → A, C → B, C → D

Canonical cover [Example]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Consider the relation schema R = (A, B, C, D, E, F) with FDs

  F = {B → A, AD → BC, C → ABD}
- Find canonical cover.

- Decompose RHS: B → A, AD → B, AD → C, C → A, C → B, C → D
- Find the closure of each FD by including and removing that FD
  - Considering B → A
    - Including **B^+ = {BA}** B → A, AD → B, AD → C, C → A, C → B, C → D **Not Equivalent**
    - Excluding **B^+ = {B}** AD → B, AD → C, C → A, C → B, C → D
      - **Can't remove**

Canonical cover [Example]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Consider the relation schema R = (A, B, C, D, E, F) with FDs

  F = {B → A, AD → BC, C → ABD}

- Find canonical cover.

- B → A, AD → B, AD → C, C → A, C → B, C → D
- Find the closure of each FD by including and removing that FD
  - Considering AD → B
    - Including AD^+ = {ADBC}   B → A, AD → B, AD → C, C → A, C → B, C → D
    - Excluding AD^+ = {ADCB}   B → A, AD → C, C → A, C → B, C → D

Canonical cover [Example]

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY (Deemed to be University)

- Consider the relation schema R = (A, B, C, D, E, F) with FDs

  F = {B → A, AD → BC, C → ABD}

- Find canonical cover.

- B → A, AD → B, AD → C, C → A, C → B, C → D
- Find the closure of each FD by including and removing that FD
  - Considering AD → B
    - Including **AD⁺ = {ADBC}** B → A, AD → B, AD → C, C → A, C → B, C → D **Equivalent**
    - Excluding **AD⁺ = {ADCB}** B → A, AD → C, C → A, C → B, C → D
      - **Remove**

      New FD

      F = {B → A, AD → C, C → A, C → B, C → D}

Canonical cover [Example]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Consider the relation schema R = (A, B, C, D, E, F) with FDs

  F = {B → A, AD → BC, C → ABD}

- Find canonical cover.

- B → A, AD → C, C → A, C → B, C → D
- Find the closure of each FD by including and removing that FD
  - Considering AD → C
    - Including **AD^+ = {ADCB}** B → A, AD → C, C → A, C → B, C → D **Not Equivalent**
    - Excluding **AD^+ = {AD}** B → A, C → A, C → B, C → D

- B → A, AD → C, C → A, C → B, C → D
  - Considering C → A
    - Including **C^+ = {CABD}** B → A, AD → C, C → A, C → B, C → D **Equivalent**
    - Excluding **C^+ = {CBDA}** B → A, AD → C, C → B, C → D
      - **Remove**

      New FD **F = {B → A, AD → C, C → B, C → D}**

Canonical cover [Example]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Consider the relation schema R = (A, B, C, D, E, F) with FDs

  F = {B → A, AD → BC, C → ABD}

- Find canonical cover.

- B → A, AD → C, C → B, C → D
  - Considering C → B
    - Including **C⁺ = {ADCB}** B → A, AD → C, C → B, C → D **Not Equivalent**
    - Excluding **C⁺ = {CD}** B → A, AD → C, C → D

- B → A, AD → C, C → B, C → D
  - Considering C → D
    - Including **C⁺ = {CABD}** B → A, AD → C, C → A, C → B, C → D **Not Equivalent**
    - Excluding **C⁺ = {CBA}** B → A, AD → C, C → A, C → B

Normal forms
1NF (First Normal Form)

1NF (First Normal Form)

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Conditions for 1NF

Each **cells of a table should contain a single value**.

- A relation R is in first normal form (1NF) if and only if it **does not contain any composite attribute or multi-valued attributes or their combinations**.

OR

- A relation R is in first normal form (1NF) if and only if **all underlying domains contain atomic values only**.

2NF (Second Normal Form)

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Conditions for 2NF

It is **in 1NF** and each **table should contain a single primary key**.

- A relation R is in second normal form (2NF)
  - if and only if it is in **1NF** and
  - **every non-primary key attribute is fully dependent on the primary key**

OR

- A relation R is in second normal form (2NF)
  - if and only if it is in **1NF** and
  - **no any non-primary key attribute is partially dependent on the primary key**

3NF (Third Normal Form)

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Conditions for 3NF

It is in 2NF and there is no transitive dependency.

(Transitive dependency???) $A \rightarrow B$ & $B \rightarrow C$ then $A \rightarrow C$

- A relation R is in third normal form (3NF)
  - if and only if it is in **2NF** and
  - **every non-key attribute is non-transitively dependent on the primary key**

OR

- A relation R is in third normal form (3NF)
  - if and only if it is in **2NF** and
  - **no any non-key attribute is transitively dependent on the primary key**

Normal forms
BCNF (Boyce-Codd Normal Form)

BCNF (Boyce-Codd Normal Form)

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Conditions for BCNF

BCNF is **based on the concept of a determinant**.

It is in **3NF** and **every determinant should be primary key**.

Primary Key

Determinant

Dependent

AccountNO → {Balance, Branch}

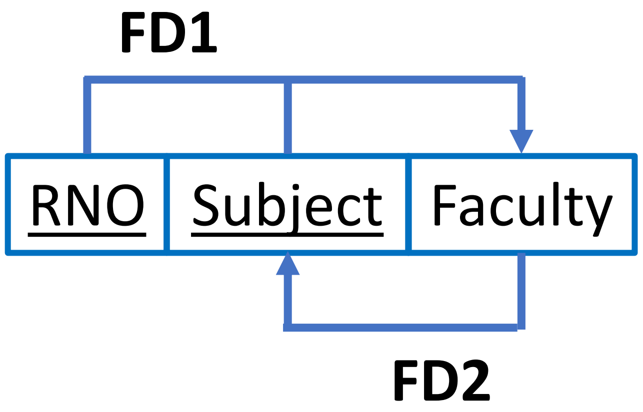
- A relation R is in Boyce-Codd normal form (BCNF)
  - if and only if it is in 3NF and
  - for every functional dependency X → Y, X should be the primary key of the table. **OR**
- A relation R is in Boyce-Codd normal form (BCNF)
  - if and only if it is in 3NF and
  - every prime key attribute is non-transitively dependent on the primary key **OR**
- A relation R is in Boyce-Codd normal form (BCNF)
  - if and only if it is in 3NF and
  - no any prime key attribute is transitively dependent on the primary key

BCNF (Boyce-Codd Normal Form) [Example]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Student

RNO	**Subject**	**Faculty**
101	DS	Patel
102	DBMS	Shah
103	DS	Jadeja
104	DBMS	Dave
105	DBMS	Shah
102	DS	Patel
101	DBMS	Dave
105	DS	Jadeja



- **FD1**: RNO, Subject → Faculty
- **FD2**: Faculty → Subject
- So {RNO, Subject} → Subject (Transitivity rule)

In FD2, **determinant is Faculty which is not a primary key**. So student table is not in BCNF.

Problem: In this relation **one student can learn more than one subject with different faculty** then **records will be stored repeatedly for each student, language and faculty combination** which **occupies more space**.

- Here, one faculty teaches only one subject, but a subject may be taught by more than one faculty.
- A student can learn a subject from only one faculty.

BCNF (Boyce-Codd Normal Form) [Example]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Student

RNO	**Subject**	**Faculty**
101	DS	Patel
102	DBMS	Shah
103	DS	Jadeja
104	DBMS	Dave
105	DBMS	Shah
102	DS	Patel
101	DBMS	Dave
105	DS	Jadeja

Table-1

Faculty	**Subject**
Patel	DS
Shah	DBMS
Jadeja	DS
Dave	DBMS

Table-2

RNO	**Faculty**
101	Patel
102	Shah
103	Jadeja
104	Dave
105	Shah
102	Patel
101	Dave
105	Jadeja

- **Solution**: Decompose relations in such a way that resultant relations do not have any transitive FD.
  - Remove the transitive dependent prime attribute from the relation that violets BCNF.
  - Place them in a separate new relation along with the non-prime attribute due to which transitive dependency occurred.
  - The primary key of the new relation will be this non-prime attribute due to which transitive dependency occurred.
  - Keep other attributes the same as in that table with the same primary key and add a prime attribute of other relation into it as a foreign key.

Normal forms and Functional Dependencies [Examples]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Normalization and FD [Example]

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY (Deemed to be University)

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

  $AB \rightarrow C$, $BC \rightarrow D$, $C \rightarrow E$

The number of super keys in the relation R is?

- **8**

Normalization and FD [Example]

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

  $A \rightarrow BC, \quad B \rightarrow CE, \quad A \rightarrow E, \quad AC \rightarrow H, \quad D \rightarrow B$

The canonical cover is?

- $A \rightarrow BH, \quad B \rightarrow CE, \quad D \rightarrow B$

Normalization and FD [Example]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

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

$$A \rightarrow BC, \quad B \rightarrow C, \quad A \rightarrow B, \quad AB \rightarrow C, \quad AC \rightarrow D$$

The canonical cover is?

- $A \rightarrow BD, \quad B \rightarrow C$

Normalization and FD [Example]

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY (Deemed to be University)

- Given the relation schema R = (A, B, C, D, E, F, G, H) and the following set of functional dependencies:

$$F = \{ A \rightarrow B,\quad ABCD \rightarrow E,\quad EF \rightarrow G,\quad EF \rightarrow H,\quad ACDF \rightarrow EG \}$$

- Compute the canonical cover and decompose R into 3NF

- $A \rightarrow B,\quad ACD \rightarrow E,\quad EF \rightarrow GH$

- R1 (A, B)
- R2 (A, C, D)
- R3 (E, F, G, H)
- R4(A, C, D, F)

Normalization and FD[Example]

- A database of research articles in a journal uses the following schema

  (VOLUME, NUMBER, STARTPAGE, ENDPAGE, TITLE, YEAR, PRICE)

The primary key is **(VOLUME, NUMBER, STARTPAGE, ENDPAGE)** and the following functional dependencies exist in the schema.

  (VOLUME, NUMBER, STARTPAGE, ENDPAGE) → TITLE

  (VOLUME, NUMBER) → YEAR

  (VOLUME, NUMBER, STARTPAGE, ENDPAGE) → PRICE

- The database is redesigned to use the following schemas.

  (VOLUME, NUMBER, STARTPAGE, ENDPAGE, TITLE, PRICE)

  (VOLUME, NUMBER, YEAR)

- Which is the weakest normal form that the new database satisfies but the old one does not?

- **2NF**

Normalization and FD[Example]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Consider the universal relation R = {A, B, C, D, E, F, G, H, I} and the set of functional dependencies

  F = { {A, B} -> {C}, {A} -> {D, E}, {B} -> {F}, {F} -> {G, H}, {D} -> {I, J} }.

- What is the key for R? Decompose R into 2NF, then 3NF relations.

- CK = {AB}
- **2NF**
  - R1 = {A, D, E, I, J} {A} -> {D, E}, {D} -> {I, J}
  - R2 = {B, F, G, H} {B} -> {F}, {F} -> {G, H}
  - R3 = {A, B, C} {A, B} -> {C}
- **3NF**
  - R11 = {A, D, E} R12 = {D, I, J}
  - R21 = {B, F} R22 = {F, G, H}
  - R3 ={A, B, C}

Normalization and FD [Example]

- Consider the relation R , which has attributes that hold schedules of courses and sections at a university;

R = { CourseNo, SecNo, OfferingDept, CreditHours, CourseLevel, InstructorSSN, Semester, Year, Days_Hours, RoomNo, NoOfStudents},

Suppose that the following functional dependencises hold on R:

{CourseNo} → {OfferngDept, CreditHours, CourseLevel}

{CourseNo, SecNo, Semester, Year} → {Days_Hours, RoomNo, NoOfStudents, InstructorSSN}

{RoomNo, Days_Hours, Semester, Year} → {InstructorSSN, CourseNo, SecNo}

- Determine which sets of attributes form keys of R. How would you normalize this relation?

- CK = {CourseNo, SecNo, Semester, Year}
- **1NF**
- Not in 2NF

Normalization and FD [Example]

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- Consider the relation R , which has attributes that hold schedules of courses and sections at a university;

R = { CourseNo, SecNo, OfferingDept, CreditHours, CourseLevel, InstructorSSN, Semester, Year, Days_Hours, RoomNo, NoOfStudents},

Suppose that the following functional dependencises hold on R:

{CourseNo} → {OfferngDept, CreditHours, CourseLevel} ***Partial Dependencies***

{CourseNo, SecNo, Semester, Year} → {Days_Hours, RoomNo, NoOfStudents, InstructorSSN}

{RoomNo, Days_Hours, Semester, Year} → {InstructorSSN, CourseNo, SecNo}

- Determine which sets of attributes form keys of R. How would you normalize this relation?

- R1 = { CourseNo, OfferingDept, CreditHours, CourseLevel}
- R2 = { CourseNo, SecNo, InstructorSSN, Semester, Year, Days_Hours, RoomNo, NoOfStudents}
- **2NF**
- 3NF?

Normalization and FD [Example]

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY (Deemed to be University)

- Consider the relation R , which has attributes that hold schedules of courses and sections at a university;

R = { CourseNo, SecNo, OfferingDept, CreditHours, CourseLevel, InstructorSSN, Semester, Year, Days_Hours, RoomNo, NoOfStudents},

Suppose that the following functional dependencises hold on R:

{CourseNo} → {OfferngDept, CreditHours, CourseLevel} ***Partial Dependencies***

{CourseNo, SecNo, Semester, Year} → {Days_Hours, RoomNo, NoOfStudents, InstructorSSN}

{RoomNo, Days_Hours, Semester, Year} → {InstructorSSN, CourseNo, SecNo}

- Determine which sets of attributes form keys of R. How would you normalize this relation?

- R1 = { <u>CourseNo</u>, OfferingDept, CreditHours, CourseLevel}
- R2 = { <u>CourseNo, SecNo</u>, InstructorSSN, <u>Semester, Year</u>, Days_Hours, RoomNo, NoOfStudents}

- **2NF**
- **3NF**
- BCNF?

Normalization and FD [Example]

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY (Deemed to be University)

- Consider the relation R , which has attributes that hold schedules of courses and sections at a university;

R = { CourseNo, SecNo, OfferingDept, CreditHours, CourseLevel, InstructorSSN, Semester, Year, Days_Hours, RoomNo, NoOfStudents},

Suppose that the following functional dependencises hold on R:

{CourseNo} → {OfferngDept, CreditHours, CourseLevel} ***Partial Dependencies***

{CourseNo, SecNo, Semester, Year} → {Days_Hours, RoomNo, NoOfStudents, InstructorSSN}

{RoomNo, Days_Hours, Semester, Year} → {InstructorSSN, CourseNo, SecNo}

- Determine which sets of attributes form keys of R. How would you normalize this relation?

- R1 = { <u>CourseNo</u>, OfferingDept, CreditHours, CourseLevel}
- R2 = { <u>CourseNo, SecNo</u>, InstructorSSN, <u>Semester, Year</u>, Days_Hours, RoomNo, NoOfStudents}
- R3 = {<u>RoomNo, Days_Hours, Semester, Year,</u> InstructorSSN, CourseNo, SecNo }

- **2NF**
- **3NF**
- **BCNF**

How to normalize database?

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- A software contract and consultancy firm maintain details of all the various projects in which its employees are currently involved. These details comprise: Employee Number, Employee Name, Date of Birth, Department Code, Department Name, Project Code, Project Description, and Project Supervisor.
- Assume the following:
  - Each employee number is unique.
  - Each department has a single department code.
  - Each project has a single code and supervisor.
  - Each employee may work on one or more projects.
  - Employee names need not necessarily be unique.
  - Project Code, Project Description, and Project Supervisor are repeating fields.
  - Normalize this data to the Third Normal Form.

How to normalize database?

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

- A software contract and consultancy firm maintain details of all the various projects in which its employees are currently involved. These details comprise: Employee Number, Employee Name, Date of Birth, Department Code, Department Name, Project Code, Project Description, Project Supervisor.

UNF

Employee Number	Employee Name	Date of Birth	Department Code	Department Name	Project Code	Project Description	Project Supervisor
1	Raj	1-1-85	1	CE	1	IOT	Patel
2	Meet	4-4-86	2	EC	2	PHP	Shah
3	Suresh	2-2-85	1	CE	1	IOT	Patel
1	Raj	1-1-85	1	CE	2	PHP	Shah

How to normalize database?

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

UNF

Employee Number	Employee Name	Date of Birth	Department Code	Department Name	Project Code	Project Description	Project Supervisor
1	Raj	1-1-85	1	CE	1	IOT	Patel
2	Meet	4-4-86	2	EC	2	PHP	Shah
3	Suresh	2-2-85	1	CE	1	IOT	Patel
1	Raj	1-1-85	1	CE	2	PHP	Shah

1NF

Employee Number	Employee Name	Date of Birth	Department Code	Department Name
1	Raj	1-1-85	1	CE
2	Meet	4-4-86	2	EC
3	Suresh	2-2-85	1	CE

Employee Number	Project Code	Project Description	Project Supervisor
1	1	IOT	Patel
2	2	PHP	Shah
3	1	IOT	Patel
1	2	PHP	Shah

How to normalize database?

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

1NF

Employee Number	Employee Name	Date of Birth	Department Code	Department Name
1	Raj	1-1-85	1	CE
2	Meet	4-4-86	2	EC
3	Suresh	2-2-85	1	CE

Employee Number	Project Code	Project Description	Project Supervisor
1	1	IOT	Patel
2	2	PHP	Shah
3	1	IOT	Patel
1	2	PHP	Shah

2NF

Employee Number	Employee Name	Date of Birth	Department Code	Department Name
1	Raj	1-1-85	1	CE
2	Meet	4-4-86	2	EC
3	Suresh	2-2-85	1	CE

Project Code	Project Description	Project Supervisor
1	IOT	Patel
2	PHP	Shah

Employee Number	Project Code
1	1
2	2
3	1
1	2

How to normalize database?

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

3NF

Employee Number	Employee Name	Date of Birth	Department Code
1	Raj	1-1-85	1
2	Meet	4-4-86	2
3	Suresh	2-2-85	1

Department Code	Department Name
1	CE
2	EC

Project Code	Project Description	Project Supervisor
1	IOT	Patel
2	PHP	Shah

Employee Number	Project Code
1	1
2	2
3	1
1	2