8.5 Third Normal Form

Definition

- A relation schema \( R \) with associated FDs \( F \) is in **third normal form (3NF)**, if, and only if, it is in 2NF and for each FD \( A \rightarrow B \in F \) at least one of the following conditions holds:
  - \( B \subseteq A \), i.e., the FD is trivial.
  - \( A \) is superkey of \( R \).
  - Each attribute \( X \in B - A \) is contained in a candidate key of \( R \).

These conditions exclude non-trivial FDs between non-key attributes. That is, transitive dependencies of the type \( A \rightarrow B \) and \( B \rightarrow C \), where \( A \) is candidate key, \( B \) is no candidate key and \( C \) contains at least one non-key attribute is forbidden.

The last condition helps to ensure that every schema has a dependency-preserving decomposition into 3NF.

Example

- relation **lecture(id, title, pers-id, room)**

- Relation is not in 3NF because the FD \( \text{pers-id} \rightarrow \text{room} \) exists, and \( \text{pers-id} \) is not a key and \( \text{room} \) is not part of a key.
Possible anomalies:
- Information about a professor and his/her room are not available without assignment of a lecture.
- Update anomaly: Change of the room number of a professor requires a change for each course with the same professor.
- Deletion anomaly: If a professor does not hold a class, all information about the professor and his/her room is removed from the database.

Solution: Splitting of the schema lecture into the two schemas lecture(id, title, pers-id) and Prof(pers-id, room).

Conclusion: The 3NF eliminates the dependencies from non-key attributes.

3NF synthesis algorithm

Goal: Decomposition of a relation schema $R$ with the FDs $F$ into relation schemas $R_1$, ..., $R_n$ so that the following three criteria are fulfilled:
- $R_1$, ..., $R_n$ is a lossless decomposition of $R$.
- The decomposition preserves the FDs.
- The schemas $R_1$, ..., $R_n$ each fulfil the 3NF.
synthesis algorithm for computing the decomposition on the basis of $F$:

- step 1: determine the canonical cover $F_c$ for $F$
  (i.e., left reduction of the FDs, right reduction of the remaining FDs, removal of
FDs of the form $A \rightarrow \emptyset$, union rule for identical left sides)

- step 2: for each FD $A \rightarrow B \in F_c$:
  + create a relation schema $R_A := A \cup B$
  + assign the FDs $F_A = \{ C \rightarrow D \in F_c \mid C \cup D \subseteq R_A \}$ to $R_A$

- step 3: If all schemas $R_A$ created in step 2 do not contain a candidate key of the
  original schema $R$, additionally create a relation with the schema $R_K = K$ and $F_K = \emptyset$
  where $K$ is candidate key of $R$.

- step 4: Eliminate schemas $R_A$ that are contained in another schema $R_A'$.

result not uniquely defined, since a set of FDs can have more than one canonical
cover, and in some cases the result of the algorithm depends on the order in which it
considers the dependencies in $F_c$
Example for decomposition algorithm

- relation schema \( \text{ProfAddr}(\text{pers-id, name, rank, room, city, street, zipcode, area-code, state, government}) \)

- assumptions:
  - A city denotes the residence of a professor.
  - Government is the party of the president.
  - City names are unique within a state.
  - The zipcode does not change within a street.
  - Cities and streets lie completely in the single states.
  - A professor has exactly one office that he does not share.

- \{\text{pers-id}\} and \{\text{room}\} are candidate keys of the relation \( \text{ProfAddr} \). The relation is not in 3NF since, e.g., the FD \{\text{city, state}\} \rightarrow \{\text{area-code}\} violates the 3NF.
8. Design Theory for Relational Databases

8.5 Third Normal Form

- step 1: computation of a canonical cover
  - FD 1: \( \{\text{pers-id}\} \rightarrow \{\text{name, rank, room, city, street, state}\} \)
  - FD 2: \( \{\text{room}\} \rightarrow \{\text{pers-id}\} \)
  - FD 3: \( \{\text{city, street, state}\} \rightarrow \{\text{zipcode}\} \)
  - FD 4: \( \{\text{city, state}\} \rightarrow \{\text{area-code}\} \)
  - FD 5: \( \{\text{state}\} \rightarrow \{\text{government}\} \)
  - FD 6: \( \{\text{zipcode}\} \rightarrow \{\text{city, state}\} \)

- step 2
  - from FD 1 we obtain:
    + \( \{\text{pers-id, name, rank, room, city, street, state}\} \)
    + FD 1 and FD 2 are assigned.
  - from FD 2 we obtain:
    + \( \{\text{room, pers-id}\} \)
    + FD 2 is assigned.
  - from FD 3 we obtain:
    + \( \{\text{city, street, state, zipcode}\} \)
    + FD 3 and FD 6 are assigned.
from FD 4 we obtain:
  + \{\text{city, state, area-code}\}
  + FD 4 is assigned.
from FD 5 we obtain:
  + \{\text{state, government}\}
  + FD 5 is assigned.
from FD 6 we obtain:
  + \{\text{zipcode, city, state}\}
  + FD 6 is assigned.

\[\text{step 3}\]

Both room and pers-id are candidate keys of the original schema ProfAddr and are contained in a schema \(R_A\).

\[\text{step 4}\]

\(\{\text{room, pers-id}\} \subseteq \{\text{pers-id, name, rank, room, city, street, state}\}\)

\(\{\text{zipcode, city, state}\} \subseteq \{\text{city, street, state, zipcode}\}\)

\[\text{We obtain:}\]
\(\{(\text{pers-id, name, rank, room, city, street, state}), \{\text{FD 1, FD 2}\}\), \(\{(\text{city, street, state, zipcode}), \{\text{FD 3, FD 6}\}\), \(\{(\text{city, state, area-code}), \{\text{FD 4}\}\), \(\{(\text{state, government}), \{\text{FD 5}\}\)\)
8.6 Boyce-Codd Normal Form

- A relation schema is in **Boyce-Codd normal form (BCNF)**, if, and only if, it is in 3NF and for all FDs $X \rightarrow \{A\}$ holds: $X$ must contain a key.

- Alternative formulation:
  A relation schema $R$ mit FDs $F$ is in BCNF, if, and only if, it is in 3NF and for each FD $A \rightarrow B \in F$ at least one of the following conditions holds:
  - $B \subseteq A$, i.e., the FD is trivial.
  - $A$ is superkey of $R$.

- Conclusion: The BCNF eliminates dependencies among attributes that are part of a key.

- Example:
  - `CarIndex(manufacturer, manufacturer-id, model-id)`
  - Consider FDs:
    + `manufacturer \rightarrow manufacturer-id (1:1-relationship)`
    + `manufacturer-id \rightarrow manufacturer`

- Example is in 3NF (all attributes are candidate keys), but not in BCNF
The following anomalies can arise:
- Insertion of the same manufacturer with different manufacturer ids (and different model ids) is possible.
- 1:1-relationship between manufacturer and manufacturer-id is connected to model-id.

properties of a schema in BCNF
- A relation schema $R$ with associated FDs $F$ can be decomposed into relation schemas $R_1, ..., R_n$ so that holds:
  + The decomposition is lossless.
  + The schemas $R_i (1 \leq i \leq n)$ are all in BCNF.
- But: We cannot always find a BCNF decomposition which is also dependency-preserving. This case is seldom in practice.

procedure
- decomposition of a schema from 3NF to BCNF
- Check if this decomposition is dependency-preserving. If this is the case, take this schema. Otherwise use the original schema in 3NF.