Normalization and Functional Dependencies

- 1NF
- Redundancy and Anomalies
- Functional Dependencies
- Attribute Closure
- Keys and Super keys
- 2NF
- 3NF
- BCNF
- Minimal Cover Algorithm
- 3NF Synthesis Algorithm
- Decomposition of Tables

1NF

- Attribute values are atomic
  - This part is assumed for any relational database
  - No repeating groups
  - Object-relational extensions to the relational model might violate 1NF depending on your definition of atomic

- Sometimes 1NF includes the requirement that a table has a primary key

Redundancy and Anomalies

- Consider combining all Library tables into one table
  - What attribute(s) could be the primary key for the table?
- Redundancy
  - The name of an author will appear in many places (once for each loan of a copy written by the author)
- Update Anomaly
  - If the customers name changes it must be changed in many places
- Delete Anomaly
  - If all loans for a copies of books written by an author are deleted all information about the author is lost. Why?
- Insert Anomaly
  - A new author cannot be added to the database until at least one loan for a copy of a book written by the author is added.
- 3NF and BCNF reduces redundancy and eliminates the anomalies described above.

Functional Dependencies

- In the following let letters late in the alphabet represent sets of attributes and letters early in the alphabet represent individual attributes
- Functional Dependencies (X -> A or X -> Y) are constraints on the data that can be entered into the database
- If the FD, X -> A, holds for a database then if t1 and t2 are tuples that contain the attributes X and attribute A (and possibly other attributes) and if the tuples have the same values for attributes X they must have the same value for attribute A

Functional Dependencies (FDs)

- FDs can entail or imply other FDs (Armstrong's Axioms)
  - Reflexivity: if Y is a subset of X then X -> Y
  - Augmentation: if X -> Y then XZ -> YZ
  - Transitivity: if X -> Y and Y -> Z then X -> Z
  - Union: if X -> A and X -> B then X -> AB
  - Decomposition: if X -> AB then X -> A and X -> B
- The closure of a set of FDs, F, is designated by F^+
- Two FD sets, F and G, are equivalent iff $F^+ = G^+$
- Equivalency of two FD sets can be shown by showing that the FDs in F are implied by the FDs in G and the FDs in G are implied by the FDs in F

Attribute Closure

- Find all attributes dependent on a particular set of attributes.
- The closure of a set of attributes, X, is designated by X^+

Attribute Closure Algorithm Under FD Set F

- *closure* := X; // since $X \subseteq X^+$

  repeat

  old := *closure*;

  if there is an FD $Z \rightarrow V$ in ***F*** such that $Z \subseteq$ *closure*

  then *closure* := *closure* $\cup$ V

  until *old* = *closure*

- If $T \subseteq$ *closure* then $X \rightarrow T$ is implied by ***F***

Problem

- Let R = {A, B, C, D, E, F}
- Let the FD set be
  - ABF → C
  - CF → B
  - CD → A
  - BD → AE
  - C → F
  - B → F
- Find the closure of ABC

Keys and Super Keys

- A set of attributes, X, in a super key for a table T if $X \subseteq T$ and X -> T
- Another way of saying this is that $T \subseteq X^+$
- A set of attributes, X, is a key for a table T if it has the super key property and no proper subset of X has the super key property

Problem

- Let R = {A, B, C, D, E, F}
- Let the FD set be
  - ABF $\rightarrow$ C
  - CF $\rightarrow$ B
  - CD $\rightarrow$ A
  - BD $\rightarrow$ AE
  - C $\rightarrow$ F
  - B $\rightarrow$ F
- Is ABF a super key for R?
- Is ABD a super key for R?
- What attribute must be part of any key for R?

2NF

- A table T is in 2NF
  - If there are no non-trivial dependencies, X -> A, that lie in T, where X is a proper subset of a key and A is not a prime attribute
- No non-prime attribute is functionally dependent on a proper subset of a key
- A prime attribute is an attribute that is part of some key
- A trivial dependency is a dependency where the right side is a subset of the left hand side
- Sometimes this is phrased as no partial key dependencies exists in the table

3NF

- A table T is in 3NF
  - if for all non-trivial dependencies, X -> A, that lie in T, X is a super key or A is a prime attribute
- An FD is a 3NF violator for table T
  - if it is a non-trivial dependency, X -> A, that lies in the T where X is not a super key and A is not a prime attribute.
-

BCNF

- A table T is in BCNF
  - if for all non-trivial dependencies, X -> A, that lie in T, X is a super key
- An FD is a BCNF violator for table T
  - if it is a non-trivial dependency, X -> A, that lies in T where X is not a super key.

Create 3NF Tables

- Identify all attributes, R, and FDs, F
  - A table containing all attributes in R is called the universal table
  - The designers must work with the customers to identify R and F
  - The FDs in F represent “real world” constraints of the data that can be entered into the database
- Create a minimal cover FD set, G, from F
- Apply the 3NF synthesis algorithm using the FD set G and the set of attributes R

Minimal Cover Set

- A minimal cover set, G, of an FD set F is an FD set such that
  - G is equivalent to F
  - No FD can be removed from G to create a "smaller" FD set equivalent to F
  - No FD in G can have an attribute removed from the FD to create a "smaller" FD set equivalent to F
- Minimal cover sets are not unique