Normalisation to 3NF

Database Systems Lecture 11
Natasha Alechina

In This Lecture

- Normalisation to 3NF
  - Data redundancy
  - Functional dependencies
  - Normal forms
  - First, Second, and Third Normal Forms
- For more information
  - Connolly and Begg chapter 13
  - Ullman and Widom ch.3.6.6 (2nd edition), 3.5 (3rd edition)

Redundancy and Normalisation

- Redundant data
  - Can be determined from other data in the database
  - Leads to various problems
    - INSERT anomalies
    - UPDATE anomalies
    - DELETE anomalies
- Normalisation
  - Aims to reduce data redundancy
  - Redundancy is expressed in terms of dependencies
  - Normal forms are defined that do not have certain types of dependency

First Normal Form

- In most definitions of the relational model
  - All data values should be atomic
  - This means that table entries should be single values, not sets or composite objects
- A relation is said to be in first normal form (1NF) if all data values are atomic

Normalisation to 1NF

To convert to a 1NF relation, split up any non-atomic values

Unnormalised

Module	Dept	Lecturer	Texts
M1	D1	L1	T1, T2
M2	D1	L1	T1, T3
M3	D1	L2	T4
M4	D2	L3	T1, T5
M5	D2	L4	T6

1NF

Module	Dept	Lecturer	Text
M1	D1	L1	T1
M1	D1	L1	T2
M2	D1	L1	T1
M2	D1	L1	T3
M3	D1	L2	T4
M4	D2	L3	T1
M4	D2	L3	T5
M5	D2	L4	T6

Problems in 1NF

1NF

Module	Dept	Lecturer	Text
M1	D1	L1	T1
M1	D1	L1	T2
M2	D1	L1	T1
M2	D1	L1	T3
M3	D1	L2	T4
M4	D2	L3	T1
M4	D2	L3	T5
M5	D2	L4	T6

- INSERT anomalies
  - Can't add a module with no texts
- UPDATE anomalies
  - To change lecturer for M1, we have to change two rows
- DELETE anomalies
  - If we remove M3, we remove L2 as well

Functional Dependencies

- Redundancy is often caused by a functional dependency
- A functional dependency (FD) is a link between two sets of attributes in a relation
- We can normalise a relation by removing undesirable FDs

- A set of attributes, A, functionally determines another set, B, or: there exists a functional dependency between A and B ($A \rightarrow B$), if whenever two rows of the relation have the same values for all the attributes in A, then they also have the same values for all the attributes in B.

Example

- {ID, modCode} → {First, Last, modName}
- {modCode} → {modName}
- {ID} → {First, Last}

ID	First	Last	modCode	modName
111	Joe	Bloggs	G51PRG	Programming
222	Anne	Smith	G51DBS	Databases

FDs and Normalisation

- We define a set of 'normal forms'
  - Each normal form has fewer FDs than the last
  - Since FDs represent redundancy, each normal form has less redundancy than the last

- Not all FDs cause a problem
  - We identify various sorts of FD that do
  - Each normal form removes a type of FD that is a problem
  - We will also need a way to remove FDs

Properties of FDs

- In any relation
  - The primary key FDs any set of attributes in that relation
    $K \rightarrow X$
    - K is the primary key, X is a set of attributes
  - Same for candidate keys
  - Any set of attributes is FD on itself
    $X \rightarrow X$

- Rules for FDs
  - Reflexivity: If B is a subset of A then
    $A \rightarrow B$
  - Augmentation: If $A \rightarrow B$ then
    $A \cup C \rightarrow B \cup C$
  - Transitivity:

If $A \rightarrow B$ and $B \rightarrow C$ then $A \rightarrow C$

FD Example

1NF

Module	Dept	Lecturer	Text
M1	D1	L1	T1
M1	D1	L1	T2
M2	D1	L1	T1
M2	D1	L1	T3
M3	D1	L2	T4
M4	D2	L3	T1
M4	D2	L3	T5
M5	D2	L4	T6

- The primary key is {Module, Text} so
  {Module, Text} → {Dept, Lecturer}
- 'Trivial' FDs, eg:
  {Text, Dept} → {Text}
  {Module} → {Module}
  {Dept, Lecturer} → { }

FD Example

1NF

Module	Dept	Lecturer	Text
M1	D1	L1	T1
M1	D1	L1	T2
M2	D1	L1	T1
M2	D1	L1	T3
M3	D1	L2	T4
M4	D2	L3	T1
M4	D2	L3	T5
M5	D2	L4	T6

- Other FDs are
  - {Module} → {Lecturer}
  - {Module} → {Dept}
  - {Lecturer} → {Dept}
  - These are non-trivial and determinants (left hand side of the dependency) are not keys.

Partial FDs and 2NF

- Partial FDs:
  - A FD, $A \rightarrow B$ is a partial FD, if some attribute of A can be removed and the FD still holds
  - Formally, there is some proper subset of A, $C \subset A$, such that $C \rightarrow B$
- Let us call attributes which are part of some candidate key, key attributes, and the rest non-key attributes.

Second normal form:

- A relation is in second normal form (2NF) if it is in 1NF and no non-key attribute is partially dependent on a candidate key
- In other words, no $C \rightarrow B$ where C is a strict subset of a candidate key and B is a non-key attribute.

Second Normal Form

1NF

Module	Dept	Lecturer	Text
M1	D1	L1	T1
M1	D1	L1	T2
M2	D1	L1	T1
M2	D1	L1	T3
M3	D1	L2	T4
M4	D2	L3	T1
M4	D2	L3	T5
M5	D2	L4	T6

- 1NF is not in 2NF
  - We have the FD {Module, Text} → {Lecturer, Dept}
  - But also {Module} → {Lecturer, Dept}
  - And so Lecturer and Dept are partially dependent on the primary key

Removing FDs

- Suppose we have a relation R with scheme S and the FD A → B where A ∩ B = { }
- Let C = S – (A U B)
- In other words:
  - A – attributes on the left hand side of the FD
  - B – attributes on the right hand side of the FD
  - C – all other attributes

- It turns out that we can split R into two parts:
- R1, with scheme C U A
- R2, with scheme A U B
- The original relation can be recovered as the natural join of R1 and R2:
- R = R1 NATURAL JOIN R2

1NF to 2NF – Example

1NF

Module	Dept	Lecturer	Text
M1	D1	L1	T1
M1	D1	L1	T2
M2	D1	L1	T1
M2	D1	L1	T3
M3	D1	L2	T4
M4	D2	L3	T1
M4	D2	L3	T5
M5	D2	L4	T6

2NFa

Module	Dept	Lecturer
M1	D1	L1
M2	D1	L1
M3	D1	L2
M4	D2	L3
M5	D2	L4

2NFb

Module	Text
M1	T1
M1	T2
M2	T1
M2	T3
M3	T4
M4	T1
M4	T5
M1	T6

Problems Resolved in 2NF

- Problems in 1NF
  - INSERT – Can't add a module with no texts
  - UPDATE – To change lecturer for M1, we have to change two rows
  - DELETE – If we remove M3, we remove L2 as well

- In 2NF the first two are resolved, but not the third one

2NFa

Module	Dept	Lecturer
M1	D1	L1
M2	D1	L1
M3	D1	L2
M4	D2	L3
M5	D2	L4

Problems Remaining in 2NF

- INSERT anomalies
  - Can't add lecturers who teach no modules
- UPDATE anomalies
  - To change the department for L1 we must alter two rows
- DELETE anomalies
  - If we delete M3 we delete L2 as well

2NFa

Module	Dept	Lecturer
M1	D1	L1
M2	D1	L1
M3	D1	L2
M4	D2	L3
M5	D2	L4

Transitive FDs and 3NF

- Transitive FDs:
  - A FD, $A \rightarrow C$ is a transitive FD, if there is some set B such that $A \rightarrow B$ and $B \rightarrow C$ are non-trivial FDs
  - $A \rightarrow B$ non-trivial means: B is not a subset of A
  - We have
    $A \rightarrow B \rightarrow C$
- Third normal form
  - A relation is in third normal form (3NF) if it is in 2NF and no non-key attribute is transitively dependent on a candidate key

Third Normal Form

2NFa

Module	Dept	Lecturer
M1	D1	L1
M2	D1	L1
M3	D1	L2
M4	D2	L3
M5	D2	L4

- 2NFa is not in 3NF
  - We have the FDs
    {Module} → {Lecturer}
    {Lecturer} → {Dept}
  - So there is a transitive FD from the primary key {Module} to {Dept}

2NF to 3NF – Example

2NFa

Module	Dept	Lecturer
M1	D1	L1
M2	D1	L1
M3	D1	L2
M4	D2	L3
M5	D2	L4

3NFa

Lecturer	Dept
L1	D1
L2	D1
L3	D2
L4	D2

3NFb

Module	Lecturer
M1	L1
M2	L1
M3	L2
M4	L3
M5	L4

Problems Resolved in 3NF

- Problems in 2NF
  - INSERT – Can't add lecturers who teach no modules
  - UPDATE – To change the department for L1 we must alter two rows
  - DELETE – If we delete M3 we delete L2 as well
- In 3NF all of these are resolved (for this relation – but 3NF can still have anomalies!)

3NFa

Lecturer	Dept
L1	D1
L2	D1
L3	D2
L4	D2

3NFb

Module	Lecturer
M1	L1
M2	L1
M3	L2
M4	L3
M5	L4

Normalisation and Design

- Normalisation is related to DB design
  - A database should normally be in 3NF at least
  - If your design leads to a non-3NF DB, then you might want to revise it
- When you find you have a non-3NF DB
  - Identify the FDs that are causing a problem
  - Think if they will lead to any insert, update, or delete anomalies
  - Try to remove them

Next Lecture

- More normalisation
  - Lossless decomposition; why our reduction to 2NF and 3NF is lossless
  - Boyce-Codd normal form (BCNF)
  - Higher normal forms
  - Denormalisation
- For more information
  - Connolly and Begg chapter 14
  - Ullman and Widom chapter 3.6