Normal Forms

Lecture 30

Robb T. Koether

Hampden-Sydney College

Mon, Apr 16, 2018

Outline

Normal Forms

- A normal form is a standard form.
- The standard forms are generally easier to work with and they are more efficient.
- Nonstandard forms can be inefficient (time or space) and can be error prone.

Normal Forms

- There are several levels of normal form for databases, each one built on the preceding level.
  - First Normal Form.
  - Second Normal Form.
  - Third Normal Form.
  - Boyce-Codd Normal Form.
  - Fourth Normal Form.
  - Fifth Normal Form.

Outline

First Normal Form

Definition (First Normal Form)

A relation is in first normal form (1NF) if the value of every attribute for every tuple is a single value (atomic).

- This rules out lists of values or sets of values.

First Normal Form

Not in First Normal Form

order_no	cat_no	dept	sales_person	quant	cust_no	cust_name
222	1234 3456	hardware	John Wilson	2 2	3333	Joe Smith
444	4567	lumber	Tim Gramm	2	4444	Sue Taylor
555	5678 6789	garden	David Simon	3 1	3333	Joe Smith
777	4567	lumber	Tim Gramm	2	7777	Bob Sponge
888	1234	hardware	Ben Sherman	1	4444	Sue Taylor

- In the `orders` table, one order may include multiple items.
- Nevertheless, we make a separate tuple for each item.

Example – 1NF Normalized Table

Example – 1NF Normalized Table

order_no	cat_no	dept	sales_person	quant	cust_no	cust_name
222	1234	hardware	John Wilson	2	3333	Joe Smith
222	3456	hardware	John Wilson	2	3333	Joe Smith
444	4567	lumber	Tim Gramm	2	4444	Sue Taylor
555	5678	garden	David Simon	3	3333	Joe Smith
555	6789	garden	David Simon	1	3333	Joe Smith
777	4567	lumber	Tim Gramm	2	7777	Bob Sponge
888	1234	hardware	Ben Sherman	1	4444	Sue Taylor

- To 1NF-normalize the table, make a separate tuple for each combination of values.

Outline

Functional Dependencies

Definition (Functional Dependency)

For any tuple t and for any set of attributes A, let $t[A]$ represent the values of t on the attributes in A. Then, for any two sets of attributes X and Y, Y is functionally dependent on X, denoted $X \rightarrow Y$, if

$$t_1[X] = t_2[X] \Rightarrow t_1[Y] = t_2[Y].$$

The relation $X \rightarrow Y$ is a functional dependency.

- This means that if two tuples have the same values for the attributes in X, then they must also have the same values for the attributes in Y.

Example

- Consider a relation `emp_proj_works` with the attributes
  - `ssn` – Social Security number
  - `lname` – Last name
  - `proj` – Project number
  - `proj_name` – Project name
  - `dept` – Department of the project
  - `hours` – Number of hours employee worked on project.
- Using the semantics of the situation, list all functional dependencies.

Example

- The following are a few functional dependencies in the `company` database.

$$
\begin{aligned}
\texttt{ssn} &\rightarrow \texttt{lname} \\
\texttt{ssn}, \texttt{dept} &\rightarrow \texttt{lname} \\
\texttt{ssn} &\rightarrow \texttt{lname}, \texttt{dept}, \texttt{proj}, \texttt{proj\_name} \\
\texttt{proj} &\rightarrow \texttt{proj\_name} \\
\texttt{proj} &\rightarrow \texttt{dept} \\
\texttt{proj} &\rightarrow \texttt{proj\_name}, \texttt{dept} \\
\texttt{ssn}, \texttt{proj} &\rightarrow \texttt{hours}
\end{aligned}
$$

- List all functional dependencies in the earlier example.

Superkeys and Keys

Definition (Superkey)

Let A be the set of all attributes in a relation R. Then a set $S \subseteq A$ is a superkey of R if $S \to A$.

Superkeys and Keys

Definition (Superkey)

Let A be the set of all attributes in a relation R. Then a set $S \subseteq A$ is a superkey of R if $S \to A$.

Definition (Candidate Key)

A collection of attributes K is a candidate key of a relation R if K is a superkey of R, but no proper subset of K is a superkey of R. A candidate key is also called a key.

Superkeys and Keys

Definition (Superkey)

Let A be the set of all attributes in a relation R. Then a set $S \subseteq A$ is a superkey of R if $S \rightarrow A$.

Definition (Candidate Key)

A collection of attributes K is a candidate key of a relation R if K is a superkey of R, but no proper subset of K is a superkey of R. A candidate key is also called a key.

Definition (Primary Key)

A primary key of a relation is an arbitrarily selected candidate key.

Superkeys and Keys

- A candidate key is a “minimal” superkey in the sense that if we remove any attribute from it, then it is no longer a superkey.
- What are the superkeys of the relation `emp_proj_works`?
- What are the candidate keys?
- What is the primary key? (our choice)

Outline

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$C \to D,$$
$$D \to A.$$

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$C \to D,$$
$$D \to A.$$

- Find all functional dependencies.

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$C \to D,$$
$$D \to A.$$

- Find all functional dependencies.
- Find all candidate keys.

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$C \to D,$$
$$D \to A.$$

- Find all functional dependencies.
- Find all candidate keys.
- Find, if any, all superkeys that are not candidate keys.

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$BC \to D,$$
$$CD \to A,$$
$$AD \to B.$$

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$BC \to D,$$
$$CD \to A,$$
$$AD \to B.$$

- Find all functional dependencies.

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$BC \to D,$$
$$CD \to A,$$
$$AD \to B.$$

- Find all functional dependencies.
- Find all candidate keys.

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$BC \to D,$$
$$CD \to A,$$
$$AD \to B.$$

- Find all functional dependencies.
- Find all candidate keys.
- Find, if any, all superkeys that are not candidate keys.

Outline

Full Functional Dependency

Definition (Full Functional Dependency)

A functional dependency $X \rightarrow Y$ is full if removal of any of the attributes from X renders it no longer a functional dependency. Otherwise, the functional dependency is partial.

- A full functional dependency is minimal for the set Y as a whole, but it may not be minimal for each attribute in Y.
- This is similar to the definition of a candidate key, but applied only to the set Y.
- If the primary key is a single attribute, then there can be no partial dependencies.

Primary Key

Definition (Prime Attribute)

A prime attribute of a relation R is an attribute that is a member of some key of R. All other attributes are non-prime.

- List the prime attributes in the previous example.
- List the non-prime attributes in the previous example.

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$C \to D,$$
$$D \to A.$$

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$C \to D,$$
$$D \to A.$$

- Which functional dependencies are full?

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$C \to D,$$
$$D \to A.$$

- Which functional dependencies are full?
- Find all prime attributes.

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$BC \to D,$$
$$CD \to A,$$
$$AD \to B.$$

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$BC \to D,$$
$$CD \to A,$$
$$AD \to B.$$

- Which functional dependencies are full?

Example

Example

Superkeys

- Let the schema be $R(A, B, C, D)$ and let the functional dependencies be

$$AB \to C,$$
$$BC \to D,$$
$$CD \to A,$$
$$AD \to B.$$

- Which functional dependencies are full?
- Find all prime attributes

Second Normal Form

Definition (Second Normal Form)

A relation is in second normal form (2NF) if every nonprime attribute A is fully functionally dependent on the primary key.

- The idea is to avoid any subsets of the primary key that could serve as a primary key for some of the nonprime attributes.

Example

Example

order_no	*cat_no*	dept	sales_person	quant	cust_no	cust_name
222	1234	hardware	John Wilson	2	3333	Joe Smith
222	3456	hardware	John Wilson	2	3333	Joe Smith
444	4567	lumber	Tim Gramm	2	4444	Sue Taylor
555	5678	garden	David Simon	3	3333	Joe Smith
555	6789	garden	David Simon	1	3333	Joe Smith
777	4567	lumber	Tim Gramm	2	7777	Bob Sponge
888	1234	hardware	Ben Sherman	1	4444	Sue Taylor

- Show that the relation is not in 2NF.

2NF Normalization

- To put a relation into 2NF, move the nonprime keys that are not fully functionally dependent on the primary key to a separate table along with that part of the primary key on which they are fully functionally dependent.

Example – 2NF Normalized Tables

Example – 2NF Normalized Tables

order_no	*cat_no*	quant
222	1234	2
222	3456	2
444	4567	2
555	5678	3
555	6789	1
777	4567	2
888	1234	1

Table 1

order_no	*dept*	sales_person
222	hardware	John Wilson
444	lumber	Tim Gramm
555	garden	David Simon
777	lumber	Tim Gramm
888	hardware	Ben Sherman

Table 2

order_no	cust_no	cust_name
222	3333	Joe Smith
444	4444	Sue Taylor
555	3333	Joe Smith
777	7777	Bob Sponge
888	4444	Sue Taylor

Table 3

- The three tables above are in 2NF.