Normalisation

Stéphane Bressan

What about this case?

$R = \{A, B, C\}$
$\Sigma = \{\{A, B\} \to \{C\}, \{C\} \to \{B\}\}$

The candidate keys are $\{A, B\}$ and $\{A, C\}$.

The functional dependency $\{C\} \to \{B\}$ is non trivial, yet $\{C\}$ is not a candidate key (notice that B is a prime attribute).

R with Σ is not in BCNF.

If we decompose into $R_1 = \{B, C\}$ with $\Sigma_1 = \{\{C\} \to \{B\}\}$ and $R_2 = \{B, C\}$ with $\Sigma_2 = \emptyset$ we lose the functional dependency $\{A, B\} \to \{C\}$.

This situation may happen when there are functional dependencies among prime attributes.

Idea of the Third Normal Form

Let us relax the BCNF requirements for prime attributes.

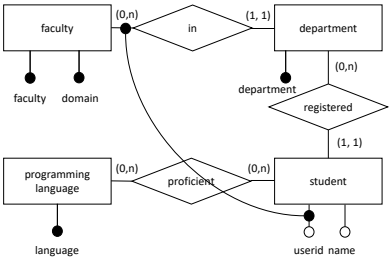
Theorem

A relation R with a set of functional dependencies Σ is in Third Normal Form, or 3NF for short, if and only if for every functional dependency $X \to \{A\} \in \Sigma^+$:

- $X \to \{A\}$ *is trivial or*
- X *is a superkey or*
- A *is a prime attribute.*

It is sufficient to look at Σ.

What if, in the case of a table `student`, there is exactly one domain for each faculty?



The strange case of a far far away dominant entity.

What if, in our case, there is exactly one domain for each faculty?

$\Sigma = \{\{userid, domain\} \rightarrow \{name, department\}, \{department\} \rightarrow \{faculty\}, \{faculty\} \rightarrow \{domain\}, \{domain\} \rightarrow \{faculty\}\}$

student				
name	userid	domain	department	faculty
Tan Hee Wee	tanh	comp.sut.edu	computer science	computing
Tan Hee Wee	tanh	comp.sut.edu	computer science	computing
Tan Hee Wee	tanh	comp.sut.edu	computer science	computing
Stanley Georgeau	stan	comp.sut.edu	computer science	computing
Goh Jin Wei	go	comp.sut.edu	information systems and analytics	computing
Tan Hee Wee	tanhw	eng.sut.edu	computer engineering	engineering
Tan Hee Wee	tanhw	eng.sut.edu	computer engineering	engineering
Bjorn Sale	bjorn	eng.sut.edu	computer engineering	engineering
Bjorn Sale	bjorn	eng.sut.edu	computer engineering	engineering
Tan Hooi Ling	tanh	sci.sut.edu	physics	science
Tan Hooi Ling	tanh	sci.sut.edu	physics	science
Roxana Nassi	rox	sci.sut.edu	mathematics	science
Amirah Mokhtar	ami	med.sut.edu	pharmacy	medecine

Let us call the table R (`student`) and the columns A (`name`) , B (`userid`), C (`domain`), D (`department`), and E (`faculty`), respectively.

We have the following compact minimal cover:
$\Sigma'' = \{\{B, C\} \rightarrow \{A, D\}, \{D\} \rightarrow \{E\}, \{C\} \rightarrow \{E\}, \{E\} \rightarrow \{C\}\}$

Calculate the attribute closures to find all the candidate keys.

...
$\{B, C\}^+ = \{A, B, C, D, E\}.$
...
$\{B, D\}^+ = \{A, B, C, D, E\}$
....
$\{B, E\}^+ = \{A, B, C, D, E\}.$

Let us call the table R (`student`) and the columns A (`name`) , B (`userid`), C (`domain`), D (`department`), and E (`faculty`), respectively.

We have the following compact minimal cover:
$\Sigma'' = \{\{B, C\} \to \{A, D\}, \{D\} \to \{E\}, \{C\} \to \{E\}, \{E\} \to \{C\}\}$

The candidate keys of R with Σ are $\{B, C\}$ (`userid`, `domain`), $\{B, E\}$ (`userid`, `faculty`), and $\{B, D\}$ (`userid`, `department`). B, C, D, and E, are prime attributes.

R with Σ is in 3NF, but not in BCNF.

$\Sigma = \{\{userid, domain\} \rightarrow \{name, department\}, \{department\} \rightarrow \{faculty\}, \{faculty\} \rightarrow \{domain\}, \{domain\} \rightarrow \{faculty\}\}$

`student` with Σ is in 3NF, but not in BCNF.

student				
name	userid	domain	department	faculty
Tan Hee Wee	tanh	comp.sut.edu	computer science	computing
Tan Hee Wee	tanh	comp.sut.edu	computer science	computing
Tan Hee Wee	tanh	comp.sut.edu	computer science	computing
Stanley Georgeau	stan	comp.sut.edu	computer science	computing
Goh Jin Wei	go	comp.sut.edu	information systems and analytics	computing
Tan Hee Wee	tanhw	eng.sut.edu	computer engineering	engineering
Tan Hee Wee	tanhw	eng.sut.edu	computer engineering	engineering
Bjorn Sale	bjorn	eng.sut.edu	computer engineering	engineering
Bjorn Sale	bjorn	eng.sut.edu	computer engineering	engineering
Tan Hooi Ling	tanh	sci.sut.edu	physics	science
Tan Hooi Ling	tanh	sci.sut.edu	physics	science
Roxana Nassi	rox	sci.sut.edu	mathematics	science
Amirah Mokhtar	ami	med.sut.edu	pharmacy	medecine

Synthesis (Bernstein Algorithm)

When a relation is not in 3NF, we can synthesise a schema in 3NF from a minimal cover of the set of functional dependencies.

- For each functional dependency $X \to Y$ in the minimal cover create a relation $R_i = X \cup Y$ unless it already exists or is subsumed by another relation.
- If none of the created relations contain one of the keys, pic a candidate key and create a relation with that candidate key.

In order to avoid unnecessary decomposition, it is generally a good idea to use a compact minimal cover (we shall do so unless we explicitly identify a problem).

The algorithm guarantees a lossless, dependency preserving decomposition in 3NF.

$R = \{A, B, C, D, E\}$
$\Sigma = \{\{A, B\} \to \{C, D, E\}, \{A, C\} \to \{B, D, E\}, \{B\} \to \{C\}, \{C\} \to \{B\}, \{C\} \to \{D\}, \{B\} \to \{E\}, \{C\} \to \{E\}\}$

Let us decompose R with Σ into a lossless dependency preserving decomposition in 3NF.

$R = \{A, B, C, D, E\}$
$\Sigma = \{\{A, B\} \to \{C, D, E\}, \{A, C\} \to \{B, D, E\}, \{B\} \to \{C\}, \{C\} \to \{B\}, \{C\} \to \{D\}, \{B\} \to \{E\}, \{C\} \to \{E\}\}$

We compute the candidate key of R with Σ.

The two candidate keys are $\{A, C\}$ and $\{A, B\}$.

We compute a compact minimal cover of R with Σ

$\Sigma'' = \{\{B\} \to \{C\}, \{C\} \to \{B, D, E\}\}$

$\Sigma'' = \{\{B\} \to \{C\}, \{C\} \to \{B, D, E\}\}$

We synthesise a relation for each functional dependency.

$R_1 = \{\underline{B}, C\}$ ($\{B\}$ is guaranteed to be candidate key of R_1 by construction).
$R_2 = \{B, \underline{C}, D, E\}$ ($\{C\}$ is guaranteed to be candidate key of R_2 by construction).

R_2 subsumes R_1. We eliminate R_1.

$R_2 = \{B, C, D, E\}$

$R_2 = \{B, C, D, E\}$

The two candidate keys are $\{A, C\}$ and $\{A, B\}$.

No relation contains one of our candidate keys of R. We add a relation with one of the candidate key.

$R_3 = \{A, C\}$ ($\{A, C\}$ is guaranteed to be a candidate key of R_3 by construction).

The resulting decomposition is:

$R_2 = \{B, C, D, E\}$ with $\Sigma_2 = \{\{B\} \to \{C\}, \{C\} \to \{B, D, E\}\}$. It is in BCNF ($\{B\}$ is also a candidate key)!

$R_3 = \{A, C\}$ with $\Sigma_3 = \emptyset$. It is in BCNF.

We could also have decomposed as:
$R_2 = \{B, C, D, E\}$.
$R_3 = \{A, B\}$.

We could also have decomposed as:
$R_1 = \{B, C\}$.
$R_2 = \{B, D, E\}$.
$R_3 = \{A, B\}$.

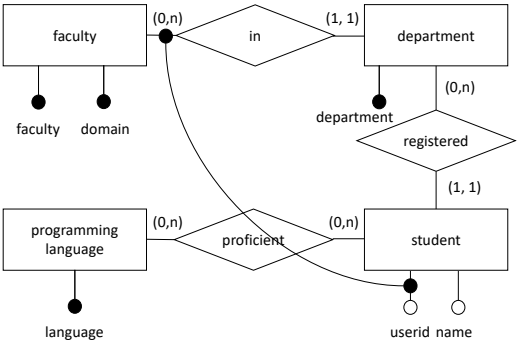
etc.

The synthesis algorithm is guaranteed to find a lossless decomposition in 3NF.

The decomposition is always dependency preserving.

Very often, the decomposition is also in BCNF.

What about our case? Let us try entity-relationship modelling and translation.



We get the following tables from the translation of the entity-relationship diagram (for instance):
$language(\underline{language})$,
$faculty(\underline{faculty}, domain)$,
$department(\underline{department}, faculty)$,
$student(\underline{userid}, faculty, name, department)$,
$proficiency(\underline{userid}, \underline{faculty}, \underline{language})$,

All tables are in BCNF except for $student$, which is in 3NF but not in BCNF.

Indeed, the projected functional dependencies on the table $student$ are $\{userid, faculty\} \to \{name, department\}$ and $\{department\} \to \{faculty\}$.
The candidate keys are $\{userid, faculty\}$ and $\{userid, department\}$.

- $\{userid, faculty\} \to \{name, department\}$: $\{userid, faculty\}$ is a superkey.
- $\{department\} \to \{faculty\}$: $faculty$ is a prime attribute.

All the functional dependencies agree with 3NF theorem: $student$ is in 3NF.

- $\{department\} \to \{faculty\}$ is non-trivial.
- $\{department\} \to \{faculty\}$: $department$ is not a superkey.

One functional dependency violates the BCNF theorem: $student$ is not in BCNF.

Let us try and decompose the table $student(userid, faculty, name, department)$ into BCNF using the violating functional dependency: $\{department\} \rightarrow \{faculty\}$.

We get the two tables:

- $R_1 = \{department, faculty\}$ with $\Sigma_1 = \{\{department\} \rightarrow \{faculty\}\}$. R_1 with Σ_1 is in BCNF.
- $R_2 = \{userid, department, name\}$ with $\Sigma_2 = \{\{userid, department\} \rightarrow \{name\}\}$ (notice the projected functional dependency). R_2 with Σ_2 is in BCNF.

But we have lost $\{userid, faculty\} \rightarrow \{department\}$ (and $\{userid, faculty\} \rightarrow \{name\}$.)

We could end up with two students having the same email!

Let us try the BCNF decomposition and the 3NF synthesis (Bernstein algorithm) on the entire table.

$\Sigma = \{\{userid, domain\} \rightarrow \{name, department\}, \{department\} \rightarrow \{faculty\}, \{faculty\} \rightarrow \{domain\}, \{domain\} \rightarrow \{faculty\}\}$

proficiency					
name	userid	domain	department	faculty	language
Tan Hee Wee	tanh	comp.sut.edu	computer science	computing	JavaScript
Tan Hee Wee	tanh	comp.sut.edu	computer science	computing	Python
Tan Hee Wee	tanh	comp.sut.edu	computer science	computing	C++
Stanley Georgeau	stan	comp.sut.edu	computer science	computing	Python
Goh Jin Wei	go	comp.sut.edu	information systems and analytics	computing	Python
Tan Hee Wee	tanhw	eng.sut.edu	computer engineering	engineering	C++
Tan Hee Wee	tanhw	eng.sut.edu	computer engineering	engineering	Fortran
Bjorn Sale	bjorn	eng.sut.edu	computer engineering	engineering	C++
Bjorn Sale	bjorn	eng.sut.edu	computer engineering	engineering	Fortran
Tan Hooi Ling	tanh	sci.sut.edu	physics	science	Julia
Tan Hooi Ling	tanh	sci.sut.edu	physics	science	Fortran
Roxana Nassi	rox	sci.sut.edu	mathematics	science	R
Amirah Mokhtar	ami	med.sut.edu	pharmacy	medecine	R

Let us call the table R (`proficiency`) and the columns A (`name`) , B (`userid`), C (`domain`), D (`department`), E (`faculty`), and F (`language`), respectively.

We have the following compact minimal cover (is it the only one?):
$\Sigma'' = \{\{B, C\} \to \{A, D\}, \{D\} \to \{E\}, \{C\} \to \{E\}, \{E\} \to \{C\}\}$

The candidate keys are $\{B, C, F\}$, $\{B, D, F\}$, and $\{B, E, F\}$.

Your turn ...

Theorem

$$(4NF) \subset BCNF \subset 3NF \subset (2NF) \subset \text{“1NF”}$$

Theorem

$$1NF \neq 2NF \neq 3NF \neq BCNF \neq 4NF$$

There are more normal forms that correspond to functional dependencies as well as other integrity constraints (e.g. multi-valued dependencies: e.g 4NF).