BCNF revisited: 40 Years Normal Forms

J.A. Makowsky
Faculty of Computer Science
Technion - IIT, Haifa

janos@cs.technion.ac.il
www.cs.technion.ac.il/∼janos

Acknowledgements

Based on work by **M.W. Vincent** and joint work with **E.V. Ravve**

See also:

[LL99] Mark Levene and George Loizou
A Guided Tour of Relational Databases and Beyond
Springer 1999

Overview

- Normal forms and functional dependencies
- BCNF and redundancy
- BCNF and update anomalies
- BCNF and storage saving
- Achieving BCNF
- Other normal forms

Functional Dependencies

$U = \{A_1, A_2, \ldots, A_m\}$ a set of attributes
F a set of functional dependencies for $R[U]$
of the form $X \rightarrow Y$ with $X, Y \subseteq U$.

A functional dependency $X \rightarrow Y$ is **trivial** if $Y \subseteq X$.

F^+ the **deductive closure of** F (with respect to the Armstrong axioms).

$K \subseteq U$ is a **superkey** for F if $K \rightarrow U \in F^+$. $K \subseteq U$ is a **key** for F if K is a superkey, but no $K' \subset K$ is a superkey.

The set of **key dependencies** of F is defined by
$F_{key} = \{K \rightarrow U \in F^+ : K \text{ is a key } \}$.

Let F be a set of functional dependencies for $R[\bar{A}, \bar{B}]$ and let $S[\bar{A}]$. We denote by $F[S]$ the set $\{X \rightarrow Y : XY \subseteq \bar{A} \text{ and } X \rightarrow \in F^+\}$, and call it the **projection of** F **on** $\bar{A}$.

Example 4.1 (from [LL99]): $EMP_1 = [ENAME, DNAME, MNAME]$

$F_1 = \{ENAME \rightarrow DNAME, DNAME \rightarrow MNAME\}$, $ENAME$ is the only key.

An instance r_1 for EMP_1 which satisfies F_1

		$EMP-1$	
	$ENAME$	$DNAME$	$MNAME$
t_1	Mark	Computing	Peter
t_2	Angela	Computing	Peter
t_3	Graham	Computing	Peter
t_4	Paul	Maths	Donald
t_5	George	Maths	Donald

We have some problems:

- We cannot add a new value for $DNAME$ without a value for $ENAME$ **Insertion Anomaly**
- We cannot delete all the values for $ENAME$ withoyt deleting all the values for $DNAME$. **Deletion Anomaly**
- It is not enough to check keys: Changing in t_1 Peter to Philip, or Computing to Maths does not violate the key. **Modification Anomaly**
- Values for $MNAME$ are repeated for every value of $ENAME$ **Redundancy Problem**

Example 4.2 (from [LL99]): $EMP_2 = [ENAME, CNAME, SAL]$

$F_2 = \{ENAME \rightarrow SAL\}$, $ENAME, CNAME$ is the only key.

An instance r_2 for EMP_2 which satisfies F_2

		$EMP-2$	
	$ENAME$	$CNAME$	SAL
t_1	Jack	Jill	25
t_2	Jack	Jake	25
t_3	Jack	John	25
t_4	Donald	Dan	30
t_5	Donald	David	30

We have the same problems:

- **Insertion Anomaly:** How to insert emplyees without children?
- **Deletion Anomaly:** How to delete children, once they are grown up?
- **Modification Anomaly:** We do not violate the key if we raise the salary from 25 to 27 only in t_1.
- **Redundancy Problem:** Salaries are repeated when employee has many children.

Example 4.3 (from [LL99]): $ADDRESS = [CITY, STREET, ZIPCODE]$

$F_3 = \{CITY, STREET \to ZIPCODE, ZIPCODE \to CITY\}$,
Both $CITY, STREET$ and $ZIPCODE, STREET$ are keys.

An instance s for $ADDRESS$ which satisfies F_3

		$ADDRESS$	
	$STREET$	$CITY$	$ZIPCODE$
t_1	Hampstead Way	London	NW11
t_2	Falloden Way	London	NW11
t_3	Oakley Gardens	London	N8
t_4	Gower Street	London	WC1E
t_5	Amhurst Rd	London	E8

Identify the problems:

- **Insertion Anomaly:** New street built...
- **Deletion Anomaly:** Zipcode deleted ... (say area is enlarged)
- **Modification Anomaly:** Change City in t_1 from London to Bristol. Keys are not violated but $ZIPCODE \to CITY$ is.
- **Redundancy Problem:** City is repeated.

Normal Forms

$(R[U], F)$ is in **Boyce-Codd Normal Form** or
$(R[U], F)$ is in **BCNF**
if $(F_{Key})^+ = F^+$.

$(R[U], F)$ is in **Third Normal Form** or $(R[U], F)$ is in **3NF**
if for every non-trivial $X \to Y \in F^+$ either

- X is a superkey or
- $Y \subset K$ for some key K for F, i.e., $K \to U \in F^+$.
  This is called a **BCNF-violation for the key** K.

Examples for Normal Forms

The relation scheme $R[CSZ]$ with
C City
S Street
Z Zipcode
and $CS \to Z, Z \to C$ is in 3NF but not in BCNF.

CS is the only key
$Z \to C$ is a BCNF-violation.

Examples for Normal Forms, II

The relation scheme $R[NSCAP]$ with

N (Name), S (Street), C (City)
A (Areacode), P (Phone number)

and $NSC \to AP$, $SC \to A$, is not in 3NF.

NSC is the only key

$R_1[NSCP]$ with $NSC \to P$, and
$R_2[SCA]$ with $SC \to A$,
are both in BCNF.

What we (should) know from the introductory course

Given a set of attributes $R[A_1, \ldots, A_m]$ and a set F of functional dependencies, we want to decompose R into a set of relations $R_1, \ldots, R_k$ which are in Normal Form such that

- **information is preserved**, i.e., for all instances $r, r_1, \ldots r_k$ which satisfy F we have that $r = r_1 \bowtie \ldots \bowtie r_k$.
- F **is preserved**, i.e., $(F[R_1] \cup \ldots \cup F[R_k])^+ = F+$.
- This can be achieved for 3NF using minimal covers.
- It cannot always be achieved for BCNF.

Why Boyce Codd Normal Form ?

- BCNF minimizes storage
- BCNF avoids redundancy
- BCNF avoids update anomalies

We have to make this precise.

How to adapt BCNF to other data models?

- Disregard the syntactic definition!
- Adapt one of the equivalent semantic definitions!
- See what you get!
- You may get different concepts for each of them!

A historic remark

1973-1980 Concepts of normal forms are developed
Consequence problem for dependencies is
recognized as central.

1980-1985 Consequence problem for dependencies is found to be **undecidable**, but for very restricted cases.
Normalforms are considered untractable......

1990- Renewed interest in normal forms emerges

2000- Normal Forms are proposed for XML.

Rferences for Normal Forms and XML

- Marcelo Arenas and Leonid Libkin
  A Normal Form for XML Documents
  ACM Transactions on Database Systems, Vol. 29, No. 1, March 2004, Pages 195-232

- Marcelo Arenas and Leonid Libkin
  An Information-Theoretic Approach to Normal Forms for Relational and XML Data
  Journal of the ACM, Vol. 52, No. 2, March 2005, pp. 246-283.

- Millist W. Vincent, Jixue Liu, and Chengfei Liu
  Strong Functional Dependencies and Their Application to Normal Forms in XML
  ACM Transactions on Database Systems, Vol. 29, No. 3, September 2004, Pages 445-462.

- Klaus-Dieter Schewe
  Redundancy, Dependencies and Normal Forms for XML Databases
  Sixteenth Australasian Database Conference (ADC2005), vol. 39 of CRPIT, ACS, pp. 7-16.

- Diem-Thu Trinh
  XML Functional Dependencies based on Tree Homomorphisms
  PhD Thesis, June 2009, Faculty of Mathematics/Informatics and Mechanical Engineering, Clausthal University of Technology, Clausthal, Germany

Redundancy, I

Let R, F be a relation scheme.

R is F**-redundant (**F^+**-redundant) on** XY if there exists a relation $r \models F$ and a non-trivial FD $X \rightarrow Y \in F$ ($\in F^+$), and at least two distinct tuples $t_1, t_2 \in r$ with $t_1[XY] = t_2[XY]$.

R is F**-redundant (**F^+**-redundant)** if there is $XY \subset U$ such that R is F-redundant (F^+-redundant) on XY.

Example: R with $F = \{A \rightarrow B, BC \rightarrow A\}$ is F-redundant, and hence F^+-redundant.

	R	
A	B	C
a_1	b_1	c_1
a_1	b_1	c_2

Redundancy, II

The set of attributes of the form XY

- with $X \rightarrow Y \in F$ and not trivial, are called **facts**.
- with $X \rightarrow Y \in F$ and not trivial, are called **explicit facts**.
- with $X \rightarrow Y \in F^+ - F$ and not trivial, are called **implicit facts**.

Observation: $R[U]$ is F-redundant (F^+-redundant) on $XY \subset U$ iff XY is a fact and XY is not a superkey.

The rationale behind redundancy is, that if R is redundant on an explicit or implicit fact XY, the fact should be stored in a different table.

R is **not** F-redundant (F^+-redundant) if every fact is a superkey.

Redundancy, III

Theorem:
(Bernstein, Goodman, 1980; M.W. Vincent 1994)

The following are equivalent:

(i) R, F is in BCNF;

(ii) R, F is not F-redundant;

(iii) R, F is not F^+-redundant;

Proof: (ii) and (iii) are equivalent by the definition of F^+.

(i) implies (ii) will be discussed on the blackboard.

(ii) implies (i) will be proven later in the lecture.

Insertion anomalies, I

We are given a relation scheme $R[U]$ and a set of FD's F with a set of candidate keys given by F_{Key}.

Let r be a relation for R with $r \models F$.

Let $t[U]$ be a tuple we want to insert.

We check whether $r \cup \{t[U]\} \models F_{Key}$.

If $r \cup \{t[U]\} \models F_{Key}$ we accept,
else we reject the insertion of $t[U]$.

If we accept, but $r \cup \{t[U]\} \not\models F$, we say that $t[U]$ is an **insertion violation**, IV.

R, F has an **insertion anomaly** if there is an r and $t[U]$, which is an insertion violation.

Insertion anomalies, Example

We look at $R[A, B, C]$ with $F = \{A \rightarrow B, B \rightarrow C\}$.

	R	
A	B	C
a_1	b_1	c_1
a_2	b_2	c_2

We want to insert (a_3, b_1, c_3).

This is compatible with $F_{Key} = \{A \rightarrow BC\}$.

	R	
A	B	C
a_1	b_1	c_1
a_2	b_2	c_2
a_3	b_1	c_3

But this violates $B \rightarrow C$.

Insertion anomalies, Theorem

Recall R, F is in BCNF iff $F_{Key} \models F$.

Theorem: (R. Fagin, 1979)
R, F is in BCNF iff
it has no insertion anomalies.

Proof:
Assume $F_{Key} \models F$, $r \models F$ and $r \cup \{t\} \models F_{Key}$.
Then $r \cup \{t\} \models F$.

The other direction needs some work and is proven later in the course.

Deletion anomalies, I

We are given a relation scheme $R[U]$ and a set of FD's F with a set of candidate keys given by F_{Key}.

Let r be a relation for R with $r \models F$.

Let $t[U] \in r$ be a tuple we want to delete.

We check whether $r - \{t[U]\} \models F_{Key}$.

If $r - \{t[U]\} \models F_{Key}$ we accept, else we reject the deletion of $t[U]$.

If we accept, but $r - \{t[U]\} \not\models F$, we say that $t[U]$ is an **deletion violation**, DV.

R, F has an **deletion anomaly** if there is an r and $t[U]$, which is an deletion violation.

Deletion anomalies, II

Observation:
Let r be a relation for R and F a set of FD's.
Let $s \subseteq r$ another relation for R.

If $r \models F$ so also $s \models F$.

Conclusion:
There are no deletion anomalies for FD's.

Note: In the presence of Multivalued Dependencies (MVD's) there may occur deletion anomalies.

Modification anomalies, I

Let r be a relation for $R[U], F$, $t \in r$, $r \models F$, K_0 be a fixed candidate key for F.

Let t' be a tuple such that $(r - \{t\}) \cup \{t'\} \models F_{Key}$ and one of the following:

(i) $t[K] = t'[K]$ for some candidate key for F;

(ii) $t[K_0] = t'[K_0]$;

(iii) $t[K] = t'[K]$ for every candidate key for F;

but $(r - \{t\}) \cup \{t'\} \not\models F$

Then r and t' show a **modification anomaly** M_i, M_{ii}, M_{iii} respectively.

Remark: Deletion anomalies can be viewed as special cases of modification anomalies.

Modification anomalies, Example

$R[ABC]$ with $F = \{A \rightarrow B, BC \rightarrow A\}$
Candidate keys AC, BC. Choose $K_0 = BC$.

	A	B	C
	a_1	b_1	c_1
t=	a_1	b_1	c_2
s=	a_2	b_2	c_2

We modify once t and once s:

	A	B	C
	a_1	b_1	c_1
t'=	a_1	b_2	c_2

$t[AC] = t'[AC]$ and F_{Key} is satisfied,
but $A \rightarrow B$ is violated.

	A	B	C
	a_1	b_1	c_1
s'=	a_1	b_2	c_2

$s[BC] = s'[BC]$ and F_{Key} is satisfied,
but $A \rightarrow B$ is violated.

In this example we cannot take care of both candidate keys simultaneously.

Modification anomalies, II

Clearly, every M_{iii} anomaly is also an M_{ii} anomaly, and every M_{ii} anomaly is also an M_i anomaly.

Observation:

If R, F is in BCNF then it has no modification anomaly M_i
(and hence neither M_{ii} and M_{iii}).

Proof: Use that $F_{key} \models F$.

Modification anomalies, III

Theorem:(M.W. Vincent, 1994)

The following are equivalent:

(i) R, F is in BCNF

(ii) R, F has no modification anomaly M_i

(iii) R, F has no modification anomaly M_{ii}

Henceforth, we speak simply of **modification anomalies**, meaning M_i-anomalies.

Remark: Vincent also introduces a normal form weaker than BCNF
but stronger than 3NF,
which is characterized by the absence of M_{iii} modification anomalies.

Relationship between anomalies

Theorem: (Theorem 4.1. in [LL99])

Let F be a set of functional dependencies over a relation scheme R.
The following are equivalent:

(i) R has an insertion anomaly with respect to F;

(ii) R is redundant with respect to F;

(iii) R has a modification anomaly with respect to F.

Proof of Theorem 4.1: (i) implies (ii)

R has an insertion anomaly given by $r \models F$ and t such that

$$r \cup \{t\} \models F_{Key} \text{ but } r \cup \{t\} \not\models F.$$

So for some $X \to A \in F^+$, where X is not a superkey, there is $t' \in r$

$$\{t\} \cup \{t'\} \not\models X \to A.$$

Let u be a tuple with $u[X_F^+] = t'[X_F^+]$ and such that for all $B \in R - X_F^+$ the value $u[B]$ does not appear in r.

Now $u \notin r$.
Since X is not a superkey, we see that R is redundant for F. Take $r' = r \cup \{u\}$ and note that $r' \models F$.

Proof of Theorem 4.1: (ii) implies (iii)

Suppose R is redundant with respect to F.

So there exist a relation over R such that $r \models F$ and for some $X \rightarrow A \in F$ there are two distinct tuples $t_1, t_2 \in r$ such that $t_1[XA] = t_2[XA]$.

Therefore $X \rightarrow A \notin F_{Key}$, and each key for R contains some attribute not in X.

Let t be a tuple over R with

$$t[X_F^+ - A] = t_1[X_F^+ - A]$$

and such that for all attributes $B \in R - (X_F^+ - A)$
$t[B]$ is a value not appearing in r.

To get the modification anomaly, we observe that

$$(r - \{t_1\}) \cup \{t\} \models F_{Key}$$

but

$$(r - \{t_1\}) \cup \{t\} \not\models F$$

Proof of Theorem 4.1: (iii) implies (i)

Suppose R has a modification anomaly.

So there is a relation r over R with $r \models F$
and tuples t, u such that

$$(r - \{u\}) \cup \{t\} \models F_{Key} \text{ but } (r - \{u\}) \cup \{t\} \not\models F.$$

Taking now $r' = r - \{u\}$ we get an insertion anomaly for r'.

Q.E.D.

End of Part I