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

Part I

- Normal forms and functional dependencies
- BCNF and redundancy
- BCNF and update anomalies

Part II

- BCNF and storage saving
- The big characterization theorem for BCNF
Guiding examples

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
\( F_{\text{key}} = \{ K \rightarrow U \in F^+ : K \text{ is a key} \} \).

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 $F_1$

<table>
<thead>
<tr>
<th>ENAME</th>
<th>DNAME</th>
<th>MNAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mark</td>
<td>Computing</td>
<td>Peter</td>
</tr>
<tr>
<td>Angela</td>
<td>Computing</td>
<td>Peter</td>
</tr>
<tr>
<td>Graham</td>
<td>Computing</td>
<td>Peter</td>
</tr>
<tr>
<td>Paul</td>
<td>Maths</td>
<td>Donald</td>
</tr>
<tr>
<td>George</td>
<td>Maths</td>
<td>Donald</td>
</tr>
</tbody>
</table>

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 $ENAME$ without 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 $F_2$

<table>
<thead>
<tr>
<th>EMP_2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ENAME</td>
<td>CNAME</td>
</tr>
<tr>
<td>$t_1$</td>
<td>Jack</td>
</tr>
<tr>
<td>$t_2$</td>
<td>Jack</td>
</tr>
<tr>
<td>$t_3$</td>
<td>Jack</td>
</tr>
<tr>
<td>$t_4$</td>
<td>Donald</td>
</tr>
<tr>
<td>$t_5$</td>
<td>Donald</td>
</tr>
</tbody>
</table>

We have the same problems:

- **Insertion Anomaly**: How to insert employees 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]):

\[ ADDRESS = [CITY, STREET, ZIPCODE] \]

\[ F_3 = \{CITY, STREET \rightarrow ZIPCODE, ZIPCODE \rightarrow CITY\} \],
Both \textit{CITY, STREET} and \textit{ZIPCODE, STREET} are keys.

An instance \( s \) for \( ADDRESS \) which satisfies \( F_3 \)

\[
\begin{array}{|c|c|c|}
\hline
\text{STREET} & \text{CITY} & \text{ZIPCODE} \\
\hline
t_1 & \text{Hampstead Way} & \text{London} & \text{NW11} \\
t_2 & \text{Falloeden Way} & \text{London} & \text{NW11} \\
t_3 & \text{Oakley Gardens} & \text{London} & \text{N8} \\
t_4 & \text{Gower Street} & \text{London} & \text{WC1E} \\
t_5 & \text{Amhurst Rd} & \text{London} & \text{E8} \\
\hline
\end{array}
\]

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 \( ZIPCODE \rightarrow CITY \) 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 \(X \rightarrow Y \in F^+\) either

- \(X\) is a superkey or

- \(Y \subseteq K\) for some key \(K\) for \(F\), i.e., \(K \rightarrow U \in F^+\).
  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 $CS \rightarrow Z, Z \rightarrow C$ is in 3NF but not in BCNF.

$CS$ is the only key
$Z \rightarrow C$ is a BCNF-violation.
Examples for Normal Forms, II

The relation scheme \( R[NSCAP] \) with

\[ \begin{align*}
N \ (\text{Name}), & \ S \ (\text{Street}), & \ C \ (\text{City}) \\
A \ (\text{Areacode}), & \ P \ (\text{Phone number})
\end{align*} \]

and \( NSC \rightarrow AP, \ SC \rightarrow A \), is not in 3NF.

\( NSC \) is the only key

\( R_1[NSCP] \) with \( NSC \rightarrow P \), and
\( R_2[SCA] \) with \( SC \rightarrow A \),
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 $r = r_1 \Join \ldots \Join r_k$.

- **$F$ 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. Normal forms are considered untractable…..

1990- Renewed interest in normal forms emerges

2000- Normal Forms are proposed for XML.
References for Normal Forms and XML

• Marcelo Arenas and Leonid Libkin
  A Normal Form for XML Documents

• Marcelo Arenas and Leonid Libkin
  An Information-Theoretic Approach to Normal Forms for Relational and XML Data

• Millist W. Vincent, Jixue Liu, and Chengfei Liu
  Strong Functional Dependencies and Their Application to Normal Forms in XML

• 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
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.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>$b_1$</td>
<td>$c_1$</td>
<td></td>
</tr>
<tr>
<td>$a_1$</td>
<td>$b_1$</td>
<td>$c_2$</td>
<td></td>
</tr>
</tbody>
</table>
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 $XY \subseteq U$ 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 $F = \{A \rightarrow B, B \rightarrow C\}$.

$$
\begin{array}{ccc}
\text{A} & \text{B} & \text{C} \\
\text{a}_1 & \text{b}_1 & \text{c}_1 \\
\text{a}_2 & \text{b}_2 & \text{c}_2 \\
\end{array}
$$

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

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

$$
\begin{array}{ccc}
\text{A} & \text{B} & \text{C} \\
\text{a}_1 & \text{b}_1 & \text{c}_1 \\
\text{a}_2 & \text{b}_2 & \text{c}_2 \\
\text{a}_3 & \text{b}_1 & \text{c}_3 \\
\end{array}
$$

But this violates $B \rightarrow C$. 
Insertion anomalies, Theorem

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

**Theorem:** (R. Fagin, 1979)

$R, F$ is in BCNF iff it has no insertion anomalies.

**Proof:**

Assume $F_{\text{Key}} \models F$, $r \models F$ and $r \cup \{t\} \models F_{\text{Key}}$.

Then $r \cup \{t\} \models F$.

The other direction needs some work and is proven later in the course.
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 $K_0 = BC$.

\[
\begin{array}{|c|c|c|}
\hline
 & A & B & C \\
\hline
 t= & a_1 & b_1 & c_1 \\
 s= & a_1 & b_1 & c_2 \\
\hline
\end{array}
\]

We modify once $t$ and once $s$:

\[
\begin{array}{|c|c|c|}
\hline
 & A & B & C \\
\hline
 t' = & a_1 & b_1 & c_1 \\
 & a_1 & b_2 & c_2 \\
\hline
\end{array}
\]

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

\[
\begin{array}{|c|c|c|}
\hline
 & A & B & C \\
\hline
 s' = & a_1 & b_1 & c_1 \\
 & a_1 & b_2 & c_2 \\
\hline
\end{array}
\]

$s[BC] = s'[BC]$ 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

$$r \cup \{t\} \models F_{\text{Key}} \text{ but } r \cup \{t\} \nvdash F.$$ 

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

$$\{t\} \cup \{t'\} \nvdash X \rightarrow A.$$ 

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 $r' \models F$. 

29
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

$$(r - \{t_1\}) \cup \{t\} \nmid F$$
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
Part II
Let $R[U], F$ be a relation scheme.
An insertion of a tuple $t$ into $r |\models F$ is said to be $F$-valid, if $r \cup \{t\} |\models F$.

A set of attributes $X \subseteq U$ is said to be unaffected by a valid insertion $r' = r \cup \{t\}$ iff $\pi_X(r) = \pi_X(r')$.

A valid insertion is $F$-unpredictable ($F^+$-unpredictable) if there exists a non-trivial $X \rightarrow Y \in F$ ($X \rightarrow Y \in F^+$) such that $XY$ is unaffected by it.
Unpredictable insertions, Example

$R[ABC]$ with $F = \{A \rightarrow B, BC \rightarrow A\}$

We look at $A \rightarrow B$:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$a_1$</td>
<td>$b_1$</td>
<td>$c_1$</td>
</tr>
</tbody>
</table>

We now insert $t$:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$a_1$</td>
<td>$b_1$</td>
<td>$c_1$</td>
</tr>
<tr>
<td></td>
<td>$a_1$</td>
<td>$b_1$</td>
<td>$c_2$</td>
</tr>
</tbody>
</table>

This is a valid insertion which does not affect $AB$. Hence it is $F$-unpredictable.

Clearly, $F$-unpredictable implies $F^+$-unpredictable.
Unpredictable insertions, II

**Observation:**
If $R, F$ has an $F^+$-unpredictable insertion, then it is not in BCNF.

**Proof:**
There is $r$ and $t$ such that $r \cup \{t\} \models F$ and hence $r \cup \{t\} \models F_{Key}$.

There is some non-trivial $X \rightarrow Y \in F^+$, and $t' \in r$ with $t \neq t'$ but $t[XY] = t'[XY]$.

Assume for contradiction, $R, F$ is in BCNF.
So $X$ is a superkey for $F$.
But $r \cup \{t\} \models F_{Key}$. So $t = t'$, a contradiction.

**Exercise:** Show that $R, F$ has a $F^+$-unpredictable insertion iff $R, F$ is $F^+$-redundant.
Theorem: (Bernstein, Goodman, 1980)
The following are equivalent:

(i) $R, F$ is in BCNF;

(ii) $R, F$ has no $F$-unpredictable insertions.

(iii) $R, F$ has no $F^+$-unpredictable insertions.
Minimizing storage, I

Let $R[U], F$ be a relation scheme, and $\pi_U R = R_i[U_i]$ be an information preserving decomposition, i.e. $F \models \forall_i R_i[U_i] = R$.

We say that the decomposition is storage saving if there are instances $r = \forall_i r_i$ such that $\sum_i |r_i| \leq |r|$.

Example:
Consider $R[ABCD]$ with
$F_1 = \{A \rightarrow BCD, C \rightarrow D\}$ (not in BCNF) and
$F_2 = \{A \rightarrow BCD, C \rightarrow A\}$ (in BCNF) and

We decompose $R$ into $R_1[ABC]$ and $R_2[CD]$ for $F_1$ and $S_1[AC]$ and $S_2[ABD]$ for $F_2$.

With $F_1$ there may be fewer values for $C$ than for $A$, but with $F_2$ this is not possible.
Observation: If $R, F$ is in BCNF then it has no storage saving decomposition.

Proposition: $R, F$ has a storage saving decomposition iff $R, F$ is $F^+$-redundant.

Proof: Assume $R, F$ is $F^+$-redundant on $XY$ with $X \rightarrow Y \in F^+$. Then there is $r \models F$ such that the decomposition $\pi_{XY} r \pi_{X(U-Y)} r$ is storage saving.

Conversely, if $R, F$ has a storage saving information preserving decomposition with $F \models \bigwedge_i R_i[U_i] = R$. So there are $X, Y \subseteq U$ and there is an $i$ such that $XY = U_i$ and $X \rightarrow Y \in F^+$. (Here we use the characterization of information preserving decompositions!)

Now it is easy to see that $R, F$ is $F^+$-redundant on $XY$. Q.E.D.
Theorem: (Biskup; Vincent and Srinivasan)

If $R, F$ is in BCNF iff it has no storage saving decomposition.

Remark: This holds also for wider dependency classes and their respective normal forms.
Relationship between anomalies (revisited)

Additionnaly to Theorem 4.1. in [LL99] we now have shown:

**Proposition:**

Let $F$ be a set of functional dependencies over a relation scheme $(R, F')$. The following are equivalent:

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

(ii) $(R, F')$ is redundant with respect to $F'$;

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

(iv) $(R, F')$ has $F'$-unpredictable insertions.

(v) $(R, F')$ has a storage saving information preserving decomposition.

Additionally, if $(R, F')$ is in BCNF, then none of the above may occur.
Completing the picture

We still need to prove the following:

**Proposition:** The following are equivalent:

(i) \((R, F)\) is **not** in BCNF;

(ii) \((R, F)\) is redundant with respect to \(F\);

**Proof:** (i) implies (ii): Suppose \((R, F)\) is not in BCNF and for some \(X \rightarrow A \in F^+\) \(X\) is not a superkey.
We take \(r\) to consist of two tuples \(t_1, t_2\) such that \(t_1[X^+] = t_2[X^+]\) and for all \(B \in U - X^+\) we have that \(t_1[B] \neq t_2[B]\).
Clearly \(r \models F\) and \((R, F)\) is redundant on \(X^+\).

(ii) implies (i): Suppose \((R, F)\) is redundant and for some \(r \models F\) and for some \(X \rightarrow A \in F^+\). But then \(X\) is not a superkey.

Q.E.D.
Characterizations of BCNF

**Theorem:** [BCNF-characterization Theorem]

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

(i) $(R, F)$ is **not** in BCNF;

(ii) $(R, F)$ has an insertion anomaly with respect to $F$;

(iii) $(R, F)$ is redundant with respect to $F$;

(iv) $(R, F)$ has a modification anomaly with respect to $F$.

(v) $(R, F)$ has $F$-unpredictable insertions.

(vi) $(R, F)$ has a storage saving information preserving decomposition.
References

• H. Mannila and K.-J. Räihä, *The design of relational databases*, Addison Wesley, 1992


References

References, II


- J.A. Makowsky and E.V. Ravve, BCNF revisited: 30 years normal forms, in preparation