

Functional Dependencies

Peter McBrien

Dept. of Computing, Imperial College London

What is wrong with this schema?

bank_data									
no	sortcode	bname	cash	type	cname	rate?	mid	amount	tdate
100	67	Strand	34005.00	current	McBrien, P.	null	1000	2300.00	1999-01-05
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1001	4000.00	1999-01-05
100	67	Strand	34005.00	current	McBrien, P.	null	1002	-223.45	1999-01-08
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1004	-100.00	1999-01-11
103	34	Goodge St	6900.67	current	Boyd, M.	null	1005	145.50	1999-01-12
100	67	Strand	34005.00	current	McBrien, P.	null	1006	10.23	1999-01-15
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1007	345.56	1999-01-15
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1008	1230.00	1999-01-15
119	56	Wimbledon	84340.45	deposit	Poulovassilis, A.	5.50	1009	5600.00	1999-01-18

```

SELECT cash
FROM bank_data
WHERE sortcode=67
  
```



cash
34005.00
34005.00
34005.00
34005.00
34005.00

What is wrong with this schema?

bank_data									
no	sortcode	bname	cash	type	cname	rate?	mid	amount	tdate
100	67	Strand	34005.00	current	McBrien, P.	null	1000	2300.00	1999-01-05
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1001	4000.00	1999-01-05
100	67	Strand	34005.00	current	McBrien, P.	null	1002	-223.45	1999-01-08
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1004	-100.00	1999-01-11
103	34	Goodge St	6900.67	current	Boyd, M.	null	1005	145.50	1999-01-12
100	67	Strand	34005.00	current	McBrien, P.	null	1006	10.23	1999-01-15
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1007	345.56	1999-01-15
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1008	1230.00	1999-01-15
119	56	Wimbledon	84340.45	deposit	Poulovassilis, A.	5.50	1009	5600.00	1999-01-18

```
SELECT DISTINCT cash
FROM bank_data
WHERE sortcode=67
```



What is wrong with this schema?

bank_data									
no	sortcode	bname	cash	type	cname	rate?	<u>mid</u>	amount	tdate
100	67	Strand	34005.00	current	McBrien, P.	null	1000	2300.00	1999-01-05
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1001	4000.00	1999-01-05
100	67	Strand	34005.00	current	McBrien, P.	null	1002	-223.45	1999-01-08
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1004	-100.00	1999-01-11
103	34	Goodge St	6900.67	current	Boyd, M.	null	1005	145.50	1999-01-12
100	67	Strand	34005.00	current	McBrien, P.	null	1006	10.23	1999-01-15
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1007	345.56	1999-01-15
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1008	1230.00	1999-01-15
119	56	Wimbledon	84340.45	deposit	Poulovassilis, A.	5.50	1009	5600.00	1999-01-18

```
SELECT DISTINCT rate
FROM bank_data
WHERE account=107
```



Problems with Updates on Redundant Data

```
INSERT INTO bank_data
VALUES (100,67, 'Strand', 33005.00, 'deposit', 'McBrien, P.', null,
       1017, -1000.00, '1999-01-21')
```

```
UPDATE bank_data
SET rate=1.00
WHERE mid=1007
```

bank_data										
no	sortcode	bname	cash	type	cname	rate?	mid	amount	tdate	
100	67	Strand	34005.00	current	McBrien, P.	null	1000	2300.00	1999-01-05	
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1001	4000.00	1999-01-05	
100	67	Strand	34005.00	current	McBrien, P.	null	1002	-223.45	1999-01-08	
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1004	-100.00	1999-01-11	
103	34	Goodge St	6900.67	current	Boyd, M.	null	1005	145.50	1999-01-12	
100	67	Strand	34005.00	current	McBrien, P.	null	1006	10.23	1999-01-15	
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	1.00	1007	345.56	1999-01-15	
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1008	1230.00	1999-01-15	
119	56	Wimbledon	84340.45	deposit	Poulovassilis, A.	5.50	1009	5600.00	1999-01-18	
100	67	Strand	33005.00	deposit	McBrien, P.	null	1017	-1000.00	1999-01-21	

```
SELECT DISTINCT cash
FROM bank_data
WHERE sortcode=67
```

cash
34005.00
33005.00

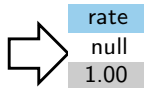
Problems with Updates on Redundant Data

```
INSERT INTO bank_data
VALUES (100,67, 'Strand', 33005.00, 'deposit', 'McBrien, P.', null,
       1017, -1000.00, '1999-01-21')
```

```
UPDATE bank_data
SET rate=1.00
WHERE mid=1007
```

bank_data									
no	sortcode	bname	cash	type	cname	rate?	mid	amount	tdate
100	67	Strand	34005.00	current	McBrien, P.	null	1000	2300.00	1999-01-05
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1001	4000.00	1999-01-05
100	67	Strand	34005.00	current	McBrien, P.	null	1002	-223.45	1999-01-08
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1004	-100.00	1999-01-11
103	34	Goodge St	6900.67	current	Boyd, M.	null	1005	145.50	1999-01-12
100	67	Strand	34005.00	current	McBrien, P.	null	1006	10.23	1999-01-15
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	1.00	1007	345.56	1999-01-15
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1008	1230.00	1999-01-15
119	56	Wimbledon	84340.45	deposit	Poulovassilis, A.	5.50	1009	5600.00	1999-01-18
100	67	Strand	33005.00	deposit	McBrien, P.	null	1017	-1000.00	1999-01-21

```
SELECT DISTINCT rate
FROM bank_data
WHERE account=107
```



How do you know what is redundant?

Functional Dependency

A **functional dependency (fd)** $X \rightarrow Y$ states that if the values of attributes X agree in two tuples, then so must the values in Y .

Using an FD to find a value

If the FD $\text{no} \rightarrow \text{rate}$ holds then x in the table below must always take the value 5.25, but y and z may take any value.

bank_data		
no	<u>mid</u>	rate
101	1001	5.25
101	1008	x
119	1009	y
z	1010	5.25

Quiz 1: FDs that hold in bank_data

bank_data									
no	sortcode	bname	cash	type	cname	rate?	<u>mid</u>	amount	tdate
100	67	Strand	34005.00	current	McBrien, P.	null	1000	2300.00	1999-01-05
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1001	4000.00	1999-01-05
100	67	Strand	34005.00	current	McBrien, P.	null	1002	-223.45	1999-01-08
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1004	-100.00	1999-01-11
103	34	Goodge St	6900.67	current	Boyd, M.	null	1005	145.50	1999-01-12
100	67	Strand	34005.00	current	McBrien, P.	null	1006	10.23	1999-01-15
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1007	345.56	1999-01-15
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1008	1230.00	1999-01-15
119	56	Wimbledon	84340.45	deposit	Poulovassilis, A.	5.50	1009	5600.00	1999-01-18

Which set of FDs below do not hold for the data?

A

no \rightarrow rate
no \rightarrow bname

B

no \rightarrow type
bname \rightarrow no

C

no \rightarrow type
mid \rightarrow bname

D

amount \rightarrow rate
bname \rightarrow sortcode

Quiz 2: Deriving FDs from other FDs

sortcode \rightarrow bname

no \rightarrow sortcode

no \rightarrow cname

no \rightarrow rate

mid \rightarrow no

Given the FDs above, which FD below might not hold?

A

no \rightarrow bname

B

no,sortcode \rightarrow cname,sortcode

C

amount,tdate \rightarrow amount

D

amount,tdate \rightarrow mid

Armstrong's Axioms

X, Y and Z are sets of attributes, and XY is a shorthand for $X \cup Y$

Reflexivity

$$Y \subseteq X \models X \rightarrow Y$$

- Such an FD is called a **trivial** FD

Applying reflexivity

If $\text{amount}, \text{tdate}$ are attributes

By reflexivity

$$\text{amount} \subseteq \text{amount}, \text{tdate} \models \text{amount}, \text{tdate} \rightarrow \text{amount}$$

$$\text{tdate} \subseteq \text{amount}, \text{tdate} \models \text{amount}, \text{tdate} \rightarrow \text{tdate}$$

Armstrong's Axioms

X, Y and Z are sets of attributes, and XY is a shorthand for $X \cup Y$

Augmentation

$$X \rightarrow Y \models XZ \rightarrow YZ$$

Applying augmentation

If $no, cname, sortcode$ are attributes and $no \rightarrow cname$

By augmentation

$$no \rightarrow cname \models no, sortcode \rightarrow cname, sortcode$$

Armstrong's Axioms

X, Y and Z are sets of attributes, and XY is a shorthand for $X \cup Y$

Transitivity

$X \rightarrow Y, Y \rightarrow Z \models X \rightarrow Z$

Applying transitivity

If $no \rightarrow \text{sortcode}$ and $\text{sortcode} \rightarrow \text{bname}$

By transitivity

$no \rightarrow \text{sortcode}, \text{sortcode} \rightarrow \text{bname} \models no \rightarrow \text{bname}$

Union Rule

Armstrong's Axioms

Reflexivity: $Y \subseteq X \models X \rightarrow Y$

Augmentation: $X \rightarrow Y \models XZ \rightarrow YZ$

Transitivity: $X \rightarrow Y, Y \rightarrow Z \models X \rightarrow Z$

Union Rule

If $X \rightarrow Y, X \rightarrow Z$

By augmentation

$X \rightarrow Y \models XZ \rightarrow YZ$

$X \rightarrow Z \models X \rightarrow XZ$

By transitivity

$X \rightarrow XZ, XZ \rightarrow YZ \models X \rightarrow YZ$

If $X \rightarrow YZ$

By reflexivity

$YZ \models YZ \rightarrow Y, YZ \rightarrow Z$

By transitivity

$X \rightarrow YZ, YZ \rightarrow Y \models X \rightarrow Y$

$X \rightarrow YZ, YZ \rightarrow Z \models X \rightarrow Z$

$\therefore X \rightarrow Y, X \rightarrow Z \equiv X \rightarrow YZ$

- Note that the union rule means that we can restrict ourselves to FD sets containing just one attribute on the RHS of each FD without losing expressiveness

Quiz 3: Deriving FDs from other FDs

Given a set $S = \{A \rightarrow BC, CD \rightarrow E, C \rightarrow F, E \rightarrow F\}$ of FDs

Which set of FDs below follows from S ?

A

$A \rightarrow BF, A \rightarrow CF, A \rightarrow ABCF$

B

$A \rightarrow BD, A \rightarrow CF, A \rightarrow ABCF$

C

$A \rightarrow BD, A \rightarrow BF, A \rightarrow ABCF$

D

$A \rightarrow BD, A \rightarrow BF, A \rightarrow CF$

Pseudotransitivity Rule

Armstrong's Axioms

Reflexivity: $Y \subseteq X \models X \rightarrow Y$

Augmentation: $X \rightarrow Y \models XZ \rightarrow YZ$

Transitivity: $X \rightarrow Y, Y \rightarrow Z \models X \rightarrow Z$

Pseudotransitivity Rule

If $X \rightarrow Y, WY \rightarrow Z$

By augmentation

$X \rightarrow Y \models WX \rightarrow WY$

By transitivity

$WX \rightarrow WY, WY \rightarrow Z \models WX \rightarrow Z$

$\therefore X \rightarrow Y, WY \rightarrow Z \models WX \rightarrow Z$

Decomposition Rule

Armstrong's Axioms

Reflexivity: $Y \subseteq X \models X \rightarrow Y$

Augmentation: $X \rightarrow Y \models XZ \rightarrow YZ$

Transitivity: $X \rightarrow Y, Y \rightarrow Z \models X \rightarrow Z$

Decomposition Rule

If $X \rightarrow Y, Z \subseteq Y$

By reflexivity

$Z \subseteq Y \models Y \rightarrow Z$

By transitivity

$X \rightarrow Y, Y \rightarrow Z \models X \rightarrow Z$

$\therefore X \rightarrow Y, Z \subseteq Y \models X \rightarrow Z$

FDs and Keys

Super-keys and minimal keys

- If a set of attributes X in relation R functionally determines all the other attributes of R , then X must be a **super-key** of R
- If it is not possible to remove any attribute from X to form X' , and X' functionally determine all attributes, then X is a **minimal key** of R

Determining keys of a relation

Suppose $\text{branch}(\text{sortcode}, \text{bname}, \text{cash})$ has the FD set $\{\text{sortcode} \rightarrow \text{bname}, \text{bname} \rightarrow \text{sortcode}, \text{bname} \rightarrow \text{cash}\}$

- 1 $\{\text{sortcode}, \text{bname}\}$ is a super-key since $\{\text{sortcode}, \text{bname}\} \rightarrow \text{cash}$
- 2 However, $\{\text{sortcode}, \text{bname}\}$ is not a minimal key, since $\text{sortcode} \rightarrow \{\text{bname}, \text{cash}\}$ and $\text{bname} \rightarrow \{\text{sortcode}, \text{cash}\}$
- 3 sortcode and bname are both minimal keys of branch

Quiz 4: Deriving minimal keys from FDs

Suppose the relation $R(A, B, C, D, E)$ has functional dependencies
 $S = \{A \rightarrow E, B \rightarrow AC, C \rightarrow D, E \rightarrow D\}$

Which of the following is a minimal key?

A

A

B

AB

C

BC

D

B

Quiz 5: Keys and FDs

Suppose the relation $R(A, B, C, D, E)$ has minimal keys AC and BC

Which FD does not necessarily hold?

A

 $ABC \rightarrow DE$

B

 $AC \rightarrow BDE$

C

 $AB \rightarrow DE$

D

 $BC \rightarrow DE$

Closure of a set of attributes with a set of FDs

Closure X^+ of a set of attributes X with FDs S

- 1 Set $X^+ := X$
- 2 Starting with X^+ apply each FD $X_s \rightarrow Y$ in S where $X_s \subseteq X^+$ but Y is not already in X^+ , to find determined attributes Y
- 3 $X^+ := X^+ \cup Y$
- 4 If Y not empty goto (2)
- 5 Return X^+

Closure of attributes

Relation $R(A, B, C, D, E, F)$ has FD set $S = \{A \rightarrow BC, CD \rightarrow E, C \rightarrow F, E \rightarrow F\}$
 To compute A^+

- Start with $A^+ = A$, just $A \rightarrow BC$ matches, so $Y = BC$
- $A^+ = ABC$, just $C \rightarrow F$ matches, so $Y = F$
- $A^+ = ABCF$, no FDs apply, so we have the result

Closure of a set of attributes with a set of FDs

Closure X^+ of a set of attributes X with FDs S

- 1 Set $X^+ := X$
- 2 Starting with X^+ apply each FD $X_s \rightarrow Y$ in S where $X_s \subseteq X^+$ but Y is not already in X^+ , to find determined attributes Y
- 3 $X^+ := X^+ \cup Y$
- 4 If Y not empty goto (2)
- 5 Return X^+

Closure of a set of attributes

Relation $R(A, B, C, D, E, F)$ has FD set $S = \{A \rightarrow BC, CD \rightarrow E, C \rightarrow F, E \rightarrow F\}$
 To compute AD^+

- Start with $AD^+ = AD$, just $A \rightarrow BC$ matches, so $Y = BC$
- $AD^+ = ABCD$, $CD \rightarrow E, C \rightarrow F$ matches, so $Y = EF$
- $AD^+ = ABCDEF$, no FDs apply, so we have the result

Quiz 6: Closure of Attribute Sets

Given a relation $R(A, B, C, D, E, F)$ and FD set
 $S = \{A \rightarrow BC, C \rightarrow D, BA \rightarrow E, BD \rightarrow F, EF \rightarrow B, BE \rightarrow ABC\}$

Which closure of attributes of S does not cover R ?

A

 A^+

B

 BC^+

C

 BE^+

D

 EF^+

Closure of a set of Functional Dependencies

Closure of the FD Set

- The closure S^+ of a set of FDs S is the set of all FDs that can be inferred from S
- Two sets of FDs S, T are equivalent if $S^+ = T^+$
- For speed, we can ignore
 - trivial FDs (*e.g.* ignore $A \rightarrow A$)
 - LHS that are not minimal (*e.g.* ignore $AB \rightarrow C$ if $A \rightarrow C$)
 - flatten all FDs to have just one attribute in RHS (*e.g.* consider $A \rightarrow CD$ as $A \rightarrow C$ and $A \rightarrow D$)
- Apart from calculating equivalence, do not normally need to compute closure

Equivalent FDs

$$S = \{A \rightarrow B, A \rightarrow C, B \rightarrow A, B \rightarrow D\}$$

$$T = \{A \rightarrow B, A \rightarrow C, A \rightarrow D, B \rightarrow A\}$$

$$S^+ = T^+ = \{A \rightarrow B, A \rightarrow C, A \rightarrow D, B \rightarrow A, B \rightarrow C, B \rightarrow D\}$$

$$\therefore S \equiv T$$

Minimal cover of a set of FDs

Minimal cover S_c of S

A minimal cover S_c of FD set S has the properties that:

- All the FDs in S can be derived from S_c (i.e. $S^+ = S_c^+$)
- It is not possible to form a new set S'_c by deleting an FD from S_c or deleting an attribute from an FD in S_c , and S'_c can still derive all the FDs in S

In general, a set of FDs may have more than one minimal cover

Deriving a minimal cover

Suppose $S = \{A \rightarrow B, BC \rightarrow A, A \rightarrow C, B \rightarrow C\}$

1 Since $B \rightarrow C$

$$BC \rightarrow A \Rightarrow B \rightarrow A$$

$$\text{Leaves } S' = \{A \rightarrow B, B \rightarrow A, A \rightarrow C, B \rightarrow C\}$$

2_a Since $A \rightarrow B, B \rightarrow C \models A \rightarrow C$

$$A \rightarrow C \Rightarrow \emptyset$$

$$\text{Leaves } S_c = \{A \rightarrow B, B \rightarrow A, B \rightarrow C\}$$

2_b Since $B \rightarrow A, A \rightarrow C \models B \rightarrow C$

$$B \rightarrow C \Rightarrow \emptyset$$

$$\text{Leaves } S_c = \{A \rightarrow B, B \rightarrow A, A \rightarrow C\}$$

Quiz 7: Minimal Cover of a Set of FDs

Given an FD set $S = \{A \rightarrow BC, C \rightarrow D, BA \rightarrow E, BD \rightarrow F, EF \rightarrow B, BE \rightarrow ABC\}$

Which is a minimal cover of S ?

A

$A \rightarrow BC, C \rightarrow D, BA \rightarrow E, BD \rightarrow F, EF \rightarrow B, BE \rightarrow ABC$

B

$A \rightarrow BC, C \rightarrow D, BA \rightarrow E, BD \rightarrow F, EF \rightarrow B, BE \rightarrow A$

C

$A \rightarrow BCE, C \rightarrow D, BD \rightarrow F, EF \rightarrow B, BE \rightarrow A$

D

$A \rightarrow BC, C \rightarrow D, B \rightarrow E, B \rightarrow F, EF \rightarrow B, BE \rightarrow A$

Worksheet: Minimal Cover

 $R(A, B, C, D, E, F, G, H)$ $S = \{AB \rightarrow DEH, BEF \rightarrow A, FGH \rightarrow C, D \rightarrow EG, EG \rightarrow BF, F \rightarrow BH\}$

- 1 Rewrite S to an equivalent set of FDs which only have a single attribute on the RHS of each FD.
- 2 Consider each FD $X \rightarrow A$, and for each $B \in X$, consider if $X \rightarrow B$ from the other FDs. If so, replace $X \rightarrow A$ by $(X - B) \rightarrow A$ in S .
- 3 Consider each FD $X \rightarrow A$, and compute X^+ without using $X \rightarrow A$. If $A \subseteq X^+$, delete $X \rightarrow A$ since it is redundant. This will give a minimal cover S_c of S .
- 4 Justify what are the minimal candidate keys of R constrained by S_c

Worksheet: Minimal Cover (Step 3)

1 $AB^+ = ABDEHGFC$

Try removing $AB \rightarrow D$: find $AB^+ = ABEH$, so can't remove.

Try removing $AB \rightarrow E$: find $AB^+ = ABDHEGFC$, so remove it from S'' to get S'''

Try removing $AB \rightarrow H$: find $AB^+ = ABDEGFHC$, so remove it from S''' to get $S'''' = \{AB \rightarrow D, EF \rightarrow A, FG \rightarrow C, D \rightarrow E, D \rightarrow G, EG \rightarrow B, EG \rightarrow F, F \rightarrow B, F \rightarrow H\}$

2 $EF^+ = EFABHDGC$

Try removing $EF \rightarrow A$: find $EF^+ = EFBH$, so can't remove.

3 $FG^+ = FGCBH$

Try removing $FG \rightarrow C$: find $FG^+ = FGBH$, so can't remove.

4 $D^+ = DEGBFHAC$

Try removing $D \rightarrow E$: find $D^+ = DG$, so can't remove.

Try removing $D \rightarrow G$: find $D^+ = DE$, so can't remove.

5 $EG^+ = EGBFHADC$

Try removing $EG \rightarrow B$: find $EG^+ = EGFBHADC$, so remove it from S'''' to get S'''''

Try removing $EG \rightarrow F$: find $EG^+ = EG$, so can't remove.

6 $F^+ = FBH$

Try removing $F \rightarrow B$: find $F^+ = FH$, so can't remove.

Try removing $F \rightarrow H$: find $F^+ = FB$, so can't remove.

Thus S''''' is a minimal cover

$$S_c = \{AB \rightarrow D, EF \rightarrow A, FG \rightarrow C, D \rightarrow EG, EG \rightarrow F, F \rightarrow BH\}$$

Normalisation

Peter McBrien

Dept. of Computing, Imperial College London

Using FDs to Formalise Problems in Schemas

bank_data									
no	sortcode	bname	cash	type	cname	rate?	mid	amount	tdate
100	67	Strand	34005.00	current	McBrien, P.	null	1000	2300.00	1999-01-05
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1001	4000.00	1999-01-05
100	67	Strand	34005.00	current	McBrien, P.	null	1002	-223.45	1999-01-08
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1004	-100.00	1999-01-11
103	34	Goodge St	6900.67	current	Boyd, M.	null	1005	145.50	1999-01-12
100	67	Strand	34005.00	current	McBrien, P.	null	1006	10.23	1999-01-15
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1007	345.56	1999-01-15
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1008	1230.00	1999-01-15
119	56	Wimbledon	84340.45	deposit	Poulovassilis, A.	5.50	1009	5600.00	1999-01-18

Formalise the intuition of redundancy by the statements of FDs

$mid \rightarrow \{tdate, amount, no\}$,

$no \rightarrow \{type, cname, rate, sortcode\}$,

$\{cname, type\} \rightarrow no$,

$sortcode \rightarrow \{bname, cash\}$

$bname \rightarrow sortcode$

1st Normal Form (1NF)

Every attribute depends on the key

Quiz 8: 1st Normal Form

bank_data										
no	sortcode	bname	cash	type	cname	rate?	<u>mid</u>	amount	tdate	
100	67	Strand	34005.00	current	McBrien, P.	null	1000	2300.00	1999-01-05	
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1001	4000.00	1999-01-05	
100	67	Strand	34005.00	current	McBrien, P.	null	1002	-223.45	1999-01-08	
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1004	-100.00	1999-01-11	
103	34	Goodge St	6900.67	current	Boyd, M.	null	1005	145.50	1999-01-12	
100	67	Strand	34005.00	current	McBrien, P.	null	1006	10.23	1999-01-15	
107	56	Wimbledon	84340.45	current	Poulovassilis, A.	null	1007	345.56	1999-01-15	
101	67	Strand	34005.00	deposit	McBrien, P.	5.25	1008	1230.00	1999-01-15	
119	56	Wimbledon	84340.45	deposit	Poulovassilis, A.	5.50	1009	5600.00	1999-01-18	

mid → {tdate, amount, no},

no → {type, cname, rate, sortcode},

{cname, type} → no,

sortcode → {bname, cash}

bname → sortcode

Is bank_data in 1st Normal form?

True

False

Prime and Non-Prime Attributes

Prime Attribute

An attribute A of relation R is **prime** if there is some minimal candidate key X of R such that $A \subseteq X$

Any other attribute $B \in Attrs(R)$ is **non-prime**

Prime and non-prime attributes of bank_data

bank_data(no,sortcode,bname,cash,type,cname,rate,mid,amount,tdate)

Has FDs $mid \rightarrow \{tdate, amount, no\}$, $no \rightarrow \{type, cname, rate, sortcode\}$,

$\{cname, type\} \rightarrow no$, $sortcode \rightarrow \{bname, cash\}$, $bname \rightarrow sortcode$

Then

- 1 the only minimal candidate key is mid
- 2 the only prime attribute is mid
- 3 non-prime attributes are no,sortcode,bname,cash,type,cname,rate,amount,tdate

3rd Normal Form (3NF)

3rd Normal Form (3NF)

For every non-trivial FD $X \rightarrow A$ on R , either

- 1 X is a super-key
- 2 A is prime

Every non-key attribute depends on the key, the whole key and nothing but the key

Failure of bank_data to meet 3NF

bank_data(no,sortcode,bname,cash,type,cname,rate,mid,amount,tdate)

- Has the following FDs where the LHS is not a super-key:
 $\text{no} \rightarrow \{\text{type}, \text{cname}, \text{rate}, \text{sortcode}\}$, $\{\text{cname}, \text{type}\} \rightarrow \text{no}$,
 $\text{sortcode} \rightarrow \{\text{bname}, \text{cash}\}$, $\text{bname} \rightarrow \text{sortcode}$
- Each of the above FD causes the relation not to meet 3NF since the RHS contains non-prime attributes

Quiz 9: Prime and nonprime attributes

Given a relation $R(A, B, C, D, E, F)$ and an FD set
 $A \rightarrow BCE, C \rightarrow D, BD \rightarrow F, EF \rightarrow B, BE \rightarrow A$

What are the nonprime attributes?

A

DEF

B

BC

C

CDF

D

CD

Quiz 10: 3rd Normal Form

Given a relation $R(A, B, C, D, E, F)$ and an FD set
 $A \rightarrow BCE, C \rightarrow D, BD \rightarrow F, EF \rightarrow B, BE \rightarrow A$

Which decomposition is not in 3NF?

A

$R_1(B, D, F), R_2(A, B, C, D, E)$

B

$R_1(A, B, C, E, F), R_2(C, D)$

C

$R_1(A, B, C, E, F), R_2(C, D), R_3(B, D, F)$

D

$R_1(B, E, F), R_2(A, C, E), R_3(C, D)$

Lossless-join decomposition of relations

Lossless-join decomposition of a Relation

A **lossless-join** decomposition of a relation R with respect to FDs S into relations R_1, \dots, R_n has the properties that:

- $Attrs(R_1) \cup \dots \cup Attrs(R_n) = Attrs(R)$
- For all possible extents of R satisfying S , $\pi_{Attrs(R_1)} R \bowtie \dots \bowtie \pi_{Attrs(R_n)} R = R$

Lossless-join decomposition of bank_data

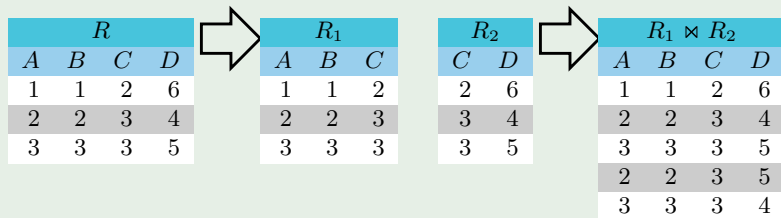
bank_data(no,sortcode,bname,cash,type,cname,rate,mid,amount,tdate)

- Has FDs $mid \rightarrow \{tdate, amount, no\}$, $no \rightarrow \{type, cname, rate, sortcode\}$, $\{cname, type\} \rightarrow no$, $sortcode \rightarrow \{bname, cash\}$, $bname \rightarrow sortcode$
- Decomposing bank_data into
 $branch = \pi_{sortcode,bname,cash} bank_data$
 $account = \pi_{no,type,cname,rate,sortcode} bank_data$
 $movement = \pi_{mid,amount,no,tdate} bank_data$
 satisfies the lossless-join decomposition property

Problems if not a lossless-join decomposition

If a decomposition of R into R_1, \dots, R_n is not lossless, then some tuples spread over R_1, \dots, R_n can result in phantom tuples appearing

$R(A, B, C, D), S = \{A \rightarrow B, B \rightarrow CD\}$



Quiz 11: Lossless join decomposition

Given a relation $R(A, B, C, D, E, F)$ and an FD set
 $A \rightarrow BCE, C \rightarrow D, BD \rightarrow F, EF \rightarrow B, BE \rightarrow A$

Which is not a lossless-join decomposition of R ?

A

$R_1(B, D, F), R_2(A, B, C, D, E)$

B

$R_1(A, B, C, E, F), R_2(C, D)$

C

$R_1(A, B, C, E, F), R_2(C, D), R_3(B, D, F)$

D

$R_1(B, E, F), R_2(A, C, E), R_3(C, D)$

Worksheet: Lossless Join Decomposition

- 1 $R(A, B, C, D, E)$ has the FDs $S = \{AB \rightarrow C, C \rightarrow DE, E \rightarrow A\}$.

Which of the following are lossless join decompositions?

1 $R_1(A, B, C), R_2(C, D, E)$

2 $R_1(A, B, C), R_2(C, D), R_3(D, E)$

- 2 Derive a lossless join decomposition into three relations of $R(A, B, C, D, E, F)$ with FDs $S = \{AB \rightarrow CD, C \rightarrow E, A \rightarrow F\}$.

- 3 Derive a lossless join decomposition into three relations of $R(A, B, C, D, E, F)$ with FDs $S = \{AB \rightarrow CD, C \rightarrow E, F \rightarrow A\}$.

Generating 3NF

Generating 3NF

- 1 Given R and a set of FDs S , find an FD $X \rightarrow A$ that causes R to violate 3NF (*i.e.* for which A is not a prime attribute and X is not a superkey).
- 2 Decompose R into $R_a(Attr(R) - A)$ and $R_b(XA)$ (Note because the two relations share X and $X \rightarrow A$ this is lossless)
- 3 Project the S onto the new relations, and repeat the process from (1)

Note that step (2) ensures that the decomposition is lossless since joining R_a with R_b will share X , and $X \rightarrow A$

Canonical Example of 3NF Decomposition

Suppose $R(A, B, C)$ has FD set $S = \{A \rightarrow B, B \rightarrow C\}$

- The only key is A , and so $B \rightarrow C$ violates 3NF (since B is not a superkey and C is nonprime).
- Decomposing R into $R_1(A, B)$ and $R_2(B, C)$ results in two 3NF relations.

Example: Decomposing bank_data into 3NF

Bank Database as a Single Relation

bank_data(no, sortcode, bname, cash, type, cname, rate, mid, amount, tdate)

$$S = \{ \text{mid} \rightarrow \{ \text{tdate}, \text{amount}, \text{no} \}, \text{no} \rightarrow \{ \text{type}, \text{cname}, \text{rate}, \text{sortcode} \}, \\ \{ \text{cname}, \text{type} \} \rightarrow \text{no}, \text{sortcode} \rightarrow \{ \text{bname}, \text{cash} \}, \text{bname} \rightarrow \text{sortcode} \}$$

Since $\text{sortcode} \rightarrow \{ \text{bname}, \text{cash} \}$ and sortcode is not superkey and $\text{bname}, \text{cash}$ nonprime, we should decompose `bank_data` into

- 1 `branch(sortcode, bname, cash)` with FDs $\text{sortcode} \rightarrow \{ \text{bname}, \text{cash} \}, \text{bname} \rightarrow \text{sortcode}$
- 2 `bank_data'(no, sortcode, type, cname, rate, mid, amount, tdate)` with FDs $\text{mid} \rightarrow \{ \text{tdate}, \text{amount}, \text{no} \}, \text{no} \rightarrow \{ \text{type}, \text{cname}, \text{rate}, \text{sortcode} \}, \{ \text{cname}, \text{type} \} \rightarrow \text{no}$

`branch` is in 3NF, but $\text{no} \rightarrow \{ \text{type}, \text{cname}, \text{rate}, \text{sortcode} \}$ makes `bank_data'` violate 3NF, so we should decompose `bank_data'` into:

- 3 `account(no, type, cname, rate, sortcode)` with FDs $\text{no} \rightarrow \{ \text{type}, \text{cname}, \text{rate}, \text{sortcode} \}, \{ \text{cname}, \text{type} \} \rightarrow \text{no}$
- 4 `movement(mid, amount, no, tdate)` with FD $\text{mid} \rightarrow \{ \text{tdate}, \text{amount}, \text{no} \}$

The relations `branch`, `account`, and `movement` are all in 3NF

Preserving FDs during decomposition

FD preserving decomposition

A lossless decomposition of R with FDs S into R_a and R_b preserves functional dependencies S if the projection of S^+ onto R_a and R_b is equivalent to S

FD preserving decomposition

Suppose $R(ABC)$ with $S = \{A \rightarrow B, B \rightarrow C, C \rightarrow A\}$ is decomposed into $R_a(AB)$ and $R_b(BC)$.

- $S^+ = \{A \rightarrow B, A \rightarrow C, B \rightarrow A, B \rightarrow C, C \rightarrow A, C \rightarrow B\}$
- The projection of S^+ onto R_a gives $S_a^+ = \{A \rightarrow B, B \rightarrow A\}$
- The projection of S^+ onto R_b gives $S_b^+ = \{B \rightarrow C, C \rightarrow B\}$
- Note that the union S_u of the two subsets of S^+ (*i.e.* $S_u = S_a^+ \cup S_b^+$) has the property that $S_u^+ = S^+$, and hence the decomposition preserves functional dependencies.

3NF

There is always possible to decompose a relation into 3NF in a manner that preserves functional dependencies. Thus any *good* 3NF decomposition of a relation must also preserve functional dependencies.

Quiz 12: Preserving FDs during Decomposition

Given a relation $R(A, B, C, D, E, F)$ and an FD set
 $A \rightarrow BCE, C \rightarrow D, BD \rightarrow F, EF \rightarrow B, BE \rightarrow A$

Which decomposition preserves FDs?

A

$R_1(B, D, F), R_2(A, B, C, D, E)$

B

$R_1(A, B, C, E, F), R_2(C, D)$

C

$R_1(A, B, C, E, F), R_2(C, D), R_3(B, D, F)$

D

$R_1(B, E, F), R_2(A, C, E), R_3(C, D)$

Preserving FDs, lossless join, and 3NF

Given a relation $R(A, B, C, D, E, F)$ and an FD set
 $A \rightarrow BCE, C \rightarrow D, BD \rightarrow F, EF \rightarrow B, BE \rightarrow A$

Decomposition	lossless join	3NF	Preserves FDs
$R_1(B, D, F), R_2(A, B, C, D, E)$	✓	✗	✗
$R_1(A, B, C, E, F), R_2(C, D)$	✓	✓	✗
$R_1(A, B, C, E, F), R_2(C, D), R_3(B, D, F)$	✓	✓	✓
$R_1(B, E, F), R_2(A, C, E), R_3(C, D)$	✗	✓	✗

Decomposing to 3NF

Since it is always possible to decompose a relation into a 3NF form that is both a lossless join decomposition, and preserves FDs, you should always do so.

Quiz 13: Preserving FDs during Decomposition to 3NF

Suppose the relation $R(A, B, C, D, E)$ has functional dependencies $S = \{AC \rightarrow DBE, BC \rightarrow DE, B \rightarrow A, E \rightarrow D\}$ (and hence has minimal keys AC and BC)

Which is a lossless join decomposition to 3NF that preserves FDs?

A

$R_a(B, C, E), R_b(A, B, C), R_c(D, E)$

B

$R_a(A, B, C), R_b(A, C, D, E)$

C

$R_a(A, C, D), R_b(A, C, E), R_c(A, B)$

D

$R_a(A, C, E), R_b(B, D, E)$

Boyce-Codd Normal Form (BCNF)

Boyce-Codd Normal Form (BCNF)

For every non-trivial FD $X \rightarrow A$ on R , X is a super-key.

Every attribute depends on the key, the whole key and nothing but the key

BCNF schema

branch(sortcode, bname, cash) with FDs $\text{sortcode} \rightarrow \{\text{bname}, \text{cash}\}$, $\text{bname} \rightarrow \text{sortcode}$ is in BCNF since sortcode and bname are both candidate keys

account(no, type, cname, rate, sortcode) with FDs $\text{no} \rightarrow \{\text{type}, \text{cname}, \text{rate}, \text{sortcode}\}$, $\{\text{cname}, \text{type}\} \rightarrow \text{no}$ is in BCNF since no and $\text{cname}, \text{type}$ are both candidate keys

movement(mid.amount, no, tdate) with FD $\text{mid} \rightarrow \{\text{tdate}, \text{amount}, \text{no}\}$ is in BCNF since mid is key

Decomposition of Relations into BCNF

Generating BCNF

- 1 Given R and a set of FDs S , find an FD $X \rightarrow A$ that causes R to violate BCNF (*i.e.* for which X is not a superkey).
- 2 Decompose R into $R_a(Attr(R) - A)$ and $R_b(XA)$ (Note because the two relations share X and $X \rightarrow A$ this is lossless)
- 3 Project the S onto the new relations, and repeat the process from (1)

Difference between 3NF and BCNF

Suppose the relation `address(no, street, town, county, postcode)` has FDs
 $\{no, street, town, county\} \rightarrow postcode$, $postcode \rightarrow \{street, town, county\}$,

- The relation is in 3NF (alternative keys `no, street, town, county` and `no, postcode`).
- The relation is not in BCNF since $postcode \rightarrow \{street, town, county\}$ has a non-superkey as the determinant
 - Decompose the relation `address` on $postcode \rightarrow \{street, town, county\}$ to:
`postcode(postcode, street, town, county)`
`streetnumber(no, postcode)`
 - Note FD $\{no, street, town, county\} \rightarrow postcode$ cannot be projected over the relations.

Worksheet: Normal Forms

$$S_c = \{AB \rightarrow D, EF \rightarrow A, FG \rightarrow C, D \rightarrow EG, EG \rightarrow F, F \rightarrow BH\}$$

- 1 Decompose the relation into 3NF
- 2 Decompose the relation into BCNF
- 3 Determine if your decompositions in (1) and (2) preserve FDs, and if they do not, suggest how to amend your schema to preserve FDs.