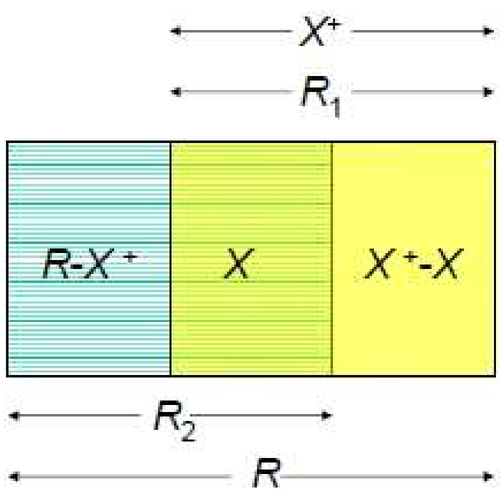
Recap: Normal Forms

- **1NF** – Every attribute is atomic.
- **2NF** – A relation is in 2NF if all non-key attributes are dependent on the whole of every candidate key.
- **3NF** – A relation is in 3NF if it is in 2NF and has no determinants except primary key.
- **BCNF** (Boyce-Codd Normal Form) – A relation is in BCNF if every determinant is a candidate key.



BCNF Decomposition: find violated $X \rightarrow B$

There are two important properties of a decomposition:

- **(1) Losslessness**:
  It should be possible to project the original relation onto the decomposed schema, and then reconstruct the original.
- **(2) Dependency Preservation**:
  It should be possible to check in the projected relations whether all the given FDs are satisfied.

BCNF gives (1), 3NF gives (1) and (2)

ICCS240 Database Management

Multivalued Dependency & Fourth Normal Form

Many slides in this lecture are either from or adapted from slides provided by

Jeff Ullman, Stanford U

Multivalued Dependencies

Multivalued dependencies (MVDs) express a condition among tuples of a relation that exists when the relation is trying to represent more than one many-many relationship.

Then certain attributes become **independent** of one another,

and their values must appear in all combinations.

Example – a new form of redundancy

Drinkers(name, addr, phones, beersLiked)

- A drinker's phones are **independent** of beers they like.
- Thus, each of a drinker's phones appears with each of the beers they like in all combinations.
- This *repetition* is unlike redundancy due to FDs, of which name $\rightarrow$ addr is the only one.

Tuples implied by Independence

If we have tuples:

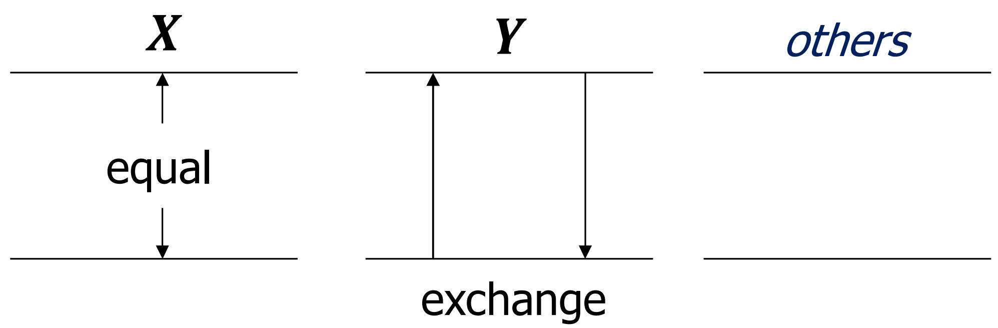
name	addr	phones	beersLiked
X	A	p1	b1
X	A	p2	b2
X	A	p2	b1
X	A	p1	b2

Then these tuples must also be in the relation.

Definition of Multivalued Dependencies

A **MVD** $X \twoheadrightarrow Y$ is an assertion that

if two tuples of a relation agree on all the attributes of X,
then their components in the set of attributes Y may be *swapped*,
and the result will be two tuples that are also in the relation.



Example

If we have tuples:

name	addr	phones	beersLiked
X	A	p1	b1
X	A	p2	b2
X	A	p2	b1
X	A	p1	b2

$$name \twoheadrightarrow phones$$
$$name \twoheadrightarrow beersLiked$$

MVD Rules

- Every FD is an MVD

- **Complementation**: If $X \twoheadrightarrow Y$, and Z is all the other attributes, then $X \twoheadrightarrow Z$.

- Like FDs, we cannot generally split the L.H.S. of an MVD.
- Unlike FDs, we cannot split the R.H.S. either.

Example of Data satisfying the MVDs

name	areaCode	phone	beersLiked	manf
Sue	10001	555-1111	Bud	Asahi
Sue	10001	555-1111	WickedAle	Heineken
Sue	10500	555-9999	Bud	Asahi
Sue	10500	555-9999	WickedAle	Heineken

But we cannot swap area codes phones by themselves.

That is, neither $\mathrm{name} \twoheadrightarrow \mathrm{areaCode}$ nor $\mathrm{name} \twoheadrightarrow \mathrm{phone}$ holds for this relation.

4NF

4NF: put MVDs into their own relation

- The redundancy that comes from MVDs is not removable by putting the database schema in BCNF.

- There is a stronger normal form, called **4NF**,
  that (intuitively) treats MVDs as FDs
  when it comes to decomposition,
  but not when determining keys of the relation.

4NF: put MVDs into their own relation

A relation R is in 4NF **if**

whenever $X \twoheadrightarrow Y$ is a non-trivial MVD,

then X is a superkey.

"**Nontrivial**" means that

- Y is not a subset of X, and
- X and Y are not, together, all the attributes.

Note that the definition of **superkey** still depends on FDs.

BCNF vs 4NF

Remember that every FD $X \rightarrow Y$ is also an MVD, $X \twoheadrightarrow Y$.

Thus, if R is in 4NF, it is certainly in BCNF.

Because any BCNF violation is a 4NF violation.

But R could be in BCNF and not 4NF

Decomposition and 4NF

If $X \twoheadrightarrow Y$ is a 4NF violation for a relation R,

we can decompose R using the same techniques as for BCNF.

- XY is one of the decomposed relations.
- All but $Y \setminus X$ is the other.

Example

Drinkers(<u>name</u>, addr, <u>phones</u>, <u>beersLiked</u>)

FDs: name $\rightarrow$ addr

MVDs: name $\twoheadrightarrow$ phones

name $\twoheadrightarrow$ beersLiked

Key is {name, phones, beersLiked}

All dependencies violate 4NF.

A relation R is in 4NF if whenever $X \twoheadrightarrow Y$ is a non-trivial MVD, then X is a superkey.

Example

Drinkers(<u>name</u>, addr, <u>phones</u>, <u>beersLiked</u>)

FDs: name → addr

MVDs: name ↠ phones

name ↠ beersLiked

Decompose using name → addr

1. Drinkers1(<u>name</u>, addr)
   in 4NF, only dependency is name → addr.
2. Drinkers2(<u>name</u>, <u>phones</u>, <u>beersLiked</u>)
   NOT in 4NF. And no FDs, so all three attributes form the key.

Example (cont.)

Continue decomposing Drinkers2

- Either MVD name ↠ phones or name ↠ beersLiked tells us to decompose to:
  Drinkers3(<u>name</u>, <u>phones</u>)
  Drinkers4(<u>name</u>, <u>beersLiked</u>)

Of note …

There are also 5NF (1979) and 6NF (2003). But they are rarely used in real life.

Practical goal:

- Redundancy = source of inconsistency/modification anomaly
- So avoid redundancy!

Exercise

Consider the following combinations of FDs on a relation $R(A, B, C, D)$:

- $AB \rightarrow C, C \rightarrow D, D \rightarrow A$
- $B \rightarrow C, B \rightarrow D$
- $AB \rightarrow C, BC \rightarrow D, CD \rightarrow A, AD \rightarrow B$
- $A \rightarrow B, B \rightarrow C, C \rightarrow D, D \rightarrow A$

For each collection of FDs do the following:

1. Indicate all the BCNF violations
2. Decompose the relations into collection of relations that are in BCNF
3. Are the decompositions dependency preserving?

Exercise

Normalize the following schema, with given constraints, to 4NF

$$
\begin{aligned}
&books(accessionno, isbn, title, author, publisher)\\
&users(userid, name, deptid, deptname)\\
&accessionno \rightarrow isbn\\
&isbn \rightarrow title\\
&isbn \rightarrow publisher\\
&isbn \rightarrow\rightarrow author\\
&userid \rightarrow name\\
&userid \rightarrow deptid\\
&deptid \rightarrow deptname
\end{aligned}
$$