Lecture 19a
The Chase Test for Lossless Join

Lossless Decomposition

- We say if a decomposition is *lossless* if the original relation can be recovered completely by natural joining the decomposed relations.
- Three important facts to remember:
  - The natural join is associative. That is, the order of the relation join does not mater.
  - Any tuple *t* in *R* is surely in the joined decomposed relations.
  - If we can prove any tuple t in joined relation R_1 *u* R_2 *u* … *u* R_k is also in *R*, we have a 1-1 mapping between *R* and re-joined relation, thus a *lossless* decomposition.

Chase Test

- The *chase* test is an organized way to see whether a tuple *t* in joined relation R_1 *u* R_2 *u* … *u* R_k can be proved using FDs also to be a tuple in *R*.
- We will show through an example how the process works.

Notations

- Assume *R* has attributes *A, B*, …
- We use *a, b,* … for the components of *t*.
- For t_i, we use the same letter as in the components that are in S_i, but we subscript the letter with *i* if the component is not in *i*.
  - The example in next slide will clarify the notation.

Example

- Suppose we have relation *R(A,B,C,D)* and FD's *A→B, B→C, CD→A*. Assume we have decomposed *R* into relations with sets of attributes S_1={A,D}, S_2={A,C}, and S_3={B,C,D}. Then the tableau (re-joined) for this decomposition looks as follows.

A	B	C	D
a	b1	c1	d
a	b2	c	d2
a3	b	c	d

Note: 1. Attribute letters with subscript mean that they are arbitrary values.
2. Attributes with subscripts are "free" values that eventually will be proved not.

Chasing 0

- Our goal is to prove those rows with subscripted attributes are really in original *R*.
- We "chase" the tableau by applying *FD*s repeatedly until the relation becomes *R*.
- In the next a few slides, we will see how the tableau evolves when *FD*s are applied

Chase 1

- Because the first two rows agree on attribute A, they must also agree on attribute B by the FD {A→B}, so the tableau evolves into

A	B	C	D
a	b1	c1	d
a	b1	c	d2
a3	b	c	d

The red colored subscript indicates the change, in this case from b2 to b1.

Chase 2

- Because the first two rows now agree on attribute B, they must also agree on attribute C by the FD {B→C}, thus c and c1 must be the same. So the tableau evolves into

A	B	C	D
a	b1	c	d
a	b1	c	d2
a3	b	c	d

The red colored subscript indicates the change, in this case c1 to c.

Chase 3

- We know {CD→A}, checking row 1 and row 3, a and a3 must agree. So the tableau evolves into

A	B	C	D
a	b1	c	d
a	b1	c	d2
a	b	c	d

The red colored subscript indicates the change, in this case a3 to a.

At this point, we see that last row becomes (a,b,c,d) which is the original R. Other rows must agree as well.

Let's revisit some early examples

Name	SSN	PhoneNumber	City
Fred	123-45-6789	206-555-1234	Seattle
Fred	123-45-6789	206-555-6543	Seattle
Joe	987-65-4321	908-555-2121	Madison
Joe	987-65-4321	908-555-1234	Madison

{SSN} → {Name,City}

This FD is *bad* because it is **not** a superkey, {SSN} can't determine {PhoneNumber}

⟹ **Not** in BCNF

10

R is decomposed into R1 and R2

Name	SSN	City
Fred	123-45-6789	Seattle
Joe	987-65-4321	Madison

SSN	PhoneNumber
123-45-6789	206-555-1234
123-45-6789	206-555-6543
987-65-4321	908-555-2121
987-65-4321	908-555-1234

{SSN} → {Name,City}

This FD is now *good* because it is the key

Now in BCNF!

R can be recovered by joining R1 and R2!

11