

CPS 116 Introduction to Database Systems

Announcements (October 13)

Midterm graded; sample solution available
Please verify your grades on Blackboard

✤ Project milestone #1 due today

Review

✤ Functional dependencies

- $X \to Y$: If two rows agree on X, they must agree on Y* A generalization of the key concept
- \blacklozenge Non-key functional dependencies: a source of redundancy
 - Non-trivial $X \to Y$ where X is not a superkey \mathcal{T} Called a BCNF violation
- * BCNF decomposition: a method for removing redundancies
 - Given R(X, Y, Z) and a BCNF violation $X \to Y$, decompose R into $R_1(X, Y)$ and $R_2(X, Z)$
 - TA lossless join decomposition
- * Schema in BCNF has no redundancy due to FD's

Next

- * 3NF (BCNF is too much)
- Multivalued dependencies: another source of redundancy
- ✤ 4NF (BCNF is not enough)

Motivation for 3NF

- ♦ Address (street_address, city, state, zip)
 - street_address, city, state \rightarrow zip
 - $zip \rightarrow city$, state
- ✤ Keys
 - {street_address, city, state}
 - {street_address, zip}
- ♦ BCNF?

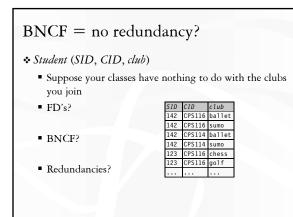
To decompose or not to decompose

Address₁ (zip, city, state) Address₂ (street_address, zip)

- ✤ FD's in Address₁
- $zip \rightarrow city$, state
- ✤ FD's in Address₂
 - None!
- ♦ Hey, where is street_address, city, state \rightarrow zip?
 - Cannot check without joining Address₁ and Address₂ back together
- Problem: Some lossless join decomposition is not dependency-preserving
- Dilemma: Should we get rid of redundancy at the expense of making constraints harder to enforce?

3NF

- ◆ *R* is in Third Normal Form (3NF) if for every non-trivial FD $X \rightarrow A$ (where *A* is single attribute), either
 - X is a superkey of R, or
 - A is a member of at least one key of R
 - ${}^{{}_{\mathrm{C}}}$ Intuitively, BCNF decomposition on $X \to A$ would "break" the key containing A
- ✤ So Address is already in 3NF
- ✤ Tradeoff:
 - Can enforce all original FD's on individual decomposed relations
 - Might have some redundancy due to FD's



Multivalued dependencies

- * A multivalued dependency (MVD) has the form $X \rightarrow Y$, where X and Y are sets of attributes in a relation R
- $X \rightarrow Y$ means that whenever two rows in R agree on all the attributes of X, then we can swap their Ycomponents and get two new rows that are also in R



MVD examples

Complete MVD + FD rules

- * FD reflexivity, augmentation, and transitivity
- ♦ MVD complementation: If $X \twoheadrightarrow Y$, then $X \twoheadrightarrow attrs(R) - X - Y$
- ♦ MVD augmentation: If $X \twoheadrightarrow Y$ and $V \subseteq W$, then $XW \twoheadrightarrow YV$
- ♦ MVD transitivity: If $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z - Y$
- - $\Rightarrow Y$ Try proving things using these!

10

11

12

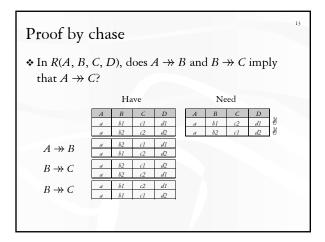
If $X \to Y$ and $Z \subseteq Y$ and there is some W disjoint from Y such that $W \to Z$, then $X \to Z$

An elegant solution: chase

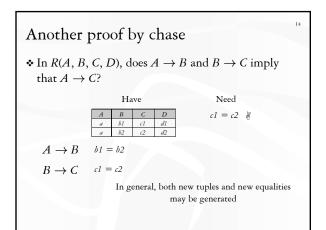
- * Given a set of FD's and MVD's \mathcal{D} , does another dependency *d* (FD or MVD) follow from \mathcal{D} ?
- * Procedure

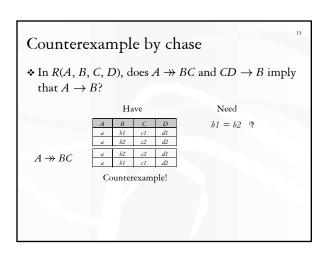
Coalescence:

- Start with the hypothesis of *d*, and treat them as "seed" tuples in a relation
- Apply the given dependencies in D repeatedly
 If we apply an FD, we infer equality of two symbols
- If we apply an MVD, we infer more tuples
- If we infer the conclusion of *d*, we have a proof
- Otherwise, if nothing more can be inferred, we have a counterexample













4NF

* A relation R is in Fourth Normal Form (4NF) if

• For every non-trivial MVD $X \twoheadrightarrow Y$ in R, X is a superkey

16

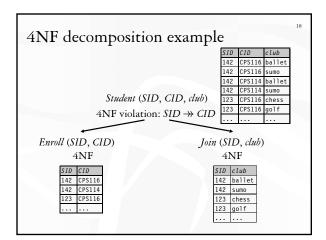
 That is, all FD's and MVD's follow from "key → other attributes" (i.e., no MVD's, and no FD's besides key functional dependencies)

* 4NF is stronger than BCNF

Because every FD is also a MVD

4NF decomposition algorithm

- Find a 4NF violation
- A non-trivial MVD $X \twoheadrightarrow Y$ in R where X is not a superkey
- Decompose R into R_1 and R_2 , where
 - $\bullet \ R_1 \text{ has attributes } X \cup Y$
 - R_2 has attributes $X \cup Z$ (Z contains attributes not in X or Y)
- \clubsuit Repeat until all relations are in 4NF
- * Almost identical to BCNF decomposition algorithm
- * Any decomposition on a 4NF violation is lossless





3NF, BCNF, 4NF, and beyond

Anomaly/normal form	3NF	BCNF	4NF
Lose FD's?	No	Possible	Possible
Redundancy due to FD's	Possible	No	No
Redundancy due to MVD's	Possible	Possible	No

* Of historical interests

- 1NF: All column values must be atomic
- 2NF: Slightly more relaxed than 3NF

Summary

Philosophy behind BCNF, 4NF: Data should depend on the key, the whole key, and nothing but the key!

20

Philosophy behind 3NF:
... But not at the expense of more expensive constraint enforcement!