Third Normal Form

Designing Schemas

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Kevin C.C. Chang, Professor Computer Science @ Illinois

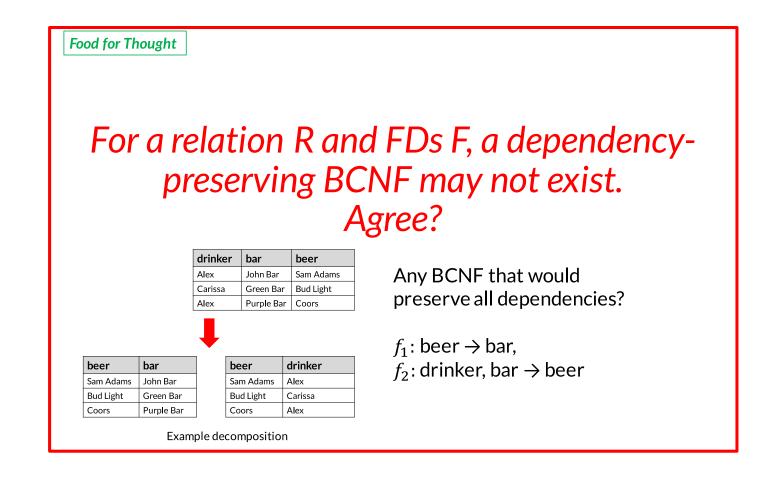
Learning Objectives

By the end of this video, you will be able to:

- Define third normal form and describe when and why it is desired.
- Determine if a given relation is in 3NF.
- Settle for 3NF-- happily-- if your BCNF decomposition does not give you a dependency-preserving schema.

Since BCNF May Not Preserve FDs

What happens if we cannot find a BCNF that preserves FDs?



3NF: Making a Compromise from BCNF

A relation R is in 3NF if and only if:

Whenever there is a **nontrivial FD** for *R*,

$$A \longrightarrow B$$

then A is a superkey for R,

or B is a prime attribute (i.e., a member of a key) for R.

Settling for 3NF

- Favorites(drinker, bar, beer)
- f_1 : beer \rightarrow bar, f_2 : drinker, bar \rightarrow beer
- It is in 3NF.
 - Key is {drinker, bar}.
 - f_1 : beer \rightarrow bar Ok, because bar is part of a key.
 - f_2 : drinker, bar \rightarrow beer Ok, because drinker, bar is a superkey.
- It is not in BCNF, though.
 - Since BCNF would not preserve FD, we will settle for 3NF.

Why Is 3NF Acceptable?

- 3NF decomposition is both:
 - Lossless decomposition
 - Dependency preserving
- It removes "bad FDs" mostly
 - Except those involving key attributes
 - I.e., will not split a key in two relations
- Favorites(drinker, bar, beer)
- f_1 : beer \rightarrow bar, f_2 : drinker, bar \rightarrow beer

drinker	bar	beer
Alex	Joe's Bar	Sam Adams
Carissa	Green Bar	Bud Light
Alex	Fancy Bar	Coors
Bob	Green Bar	Bud Light

Example Favorites relation