Boyce-Codd Normal Form

Designing Schemas

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Kevin C.C. Chang, Professor

Computer Science @ Illinois

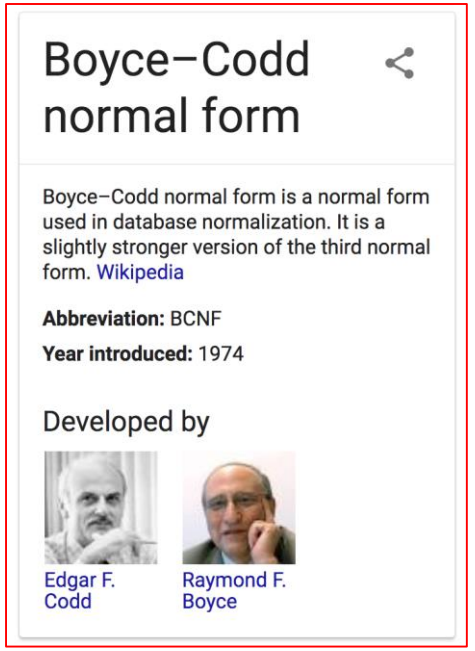
Learning Objectives

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

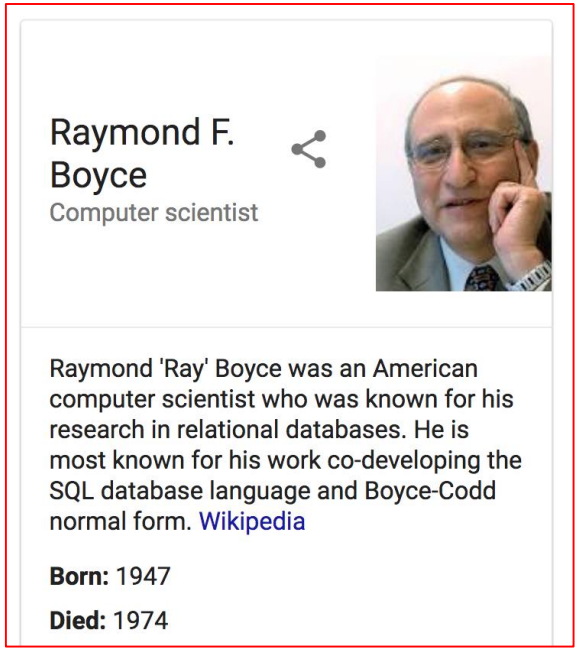
- Define what BCNF is.
- Determine whether a given relation is in BCNF.
- Transform a relation into BCNF with BCNF decomposition.

Towards BCNF: Boyce and Codd

- **Codd** invented the relational model, created the era of declarative (in contrast to procedural) data management.
- **Boyce** -- in addition to BCNF, he has a major contribution to make declarative data management a reality. *Stay tuned!!*



Google search result of ”Boyce Codd Normal Form”



Google search result of “Raymond F. Boyce”

Recall the Problematic Schema

- Why is it bad?
- A student's name and birthday are repeated for each major.
- So? We should not store name and birthday with major.

id	name	major	birthday
1	Bugs Bunny	CS	2004-11-06
1	Bugs Bunny	Music	2004-11-06
2	Donald Duck	Bio	1997-02-01
3	Peter Pan	Econ	1998-10-01
3	Peter Pan	Social	1998-10-01
3	Peter Pan	ME	1998-10-01
4	Mickey Mouse	CS	1995-04-01

Example Students table

What Is the Culprit?

- Why would name and birthday repeat, but major does not?

id	name	major	birthday
1	Bugs Bunny	CS	2004-11-06
1	Bugs Bunny	Music	2004-11-06
2	Donald Duck	Bio	1997-02-01
3	Peter Pan	Econ	1998-10-01
3	Peter Pan	Social	1998-10-01
3	Peter Pan	ME	1998-10-01
4	Mickey Mouse	CS	1995-04-01

Example Students table

- id $\longrightarrow$ name, birthday
- id $\not\longrightarrow$ major

- id determines some attributes A.
- id does not determine other attributes; i.e., it is not a superkey.
- Thus, A will repeat!
- Lesson: *An FD by non-key attributes can cause redundancy*.

Boyce-Codd Normal Form

- A relation R is in BCNF if and only if:

  Whenever there is a **nontrivial FD** for *R*,
  $$A \longrightarrow B$$
  then A is a superkey for *R*.

- Whenever a set of attributes of R determines another attribute, it should determine all attributes of R.

- That is, no bad FDs!

BCNF or Not?

- Likes(name, addr, likeBeer)
  - name $\longrightarrow$ addr
  - name $\not\longrightarrow$ likeBeer
- BCNF?

name	addr	likeBeer
Alex	100 Green St	Sam Adams
Bob	300 Purple St	Sam Adams
Carissa	200 Green St	Bud Light
Alex	100 Green St	Coors

Example Likes table

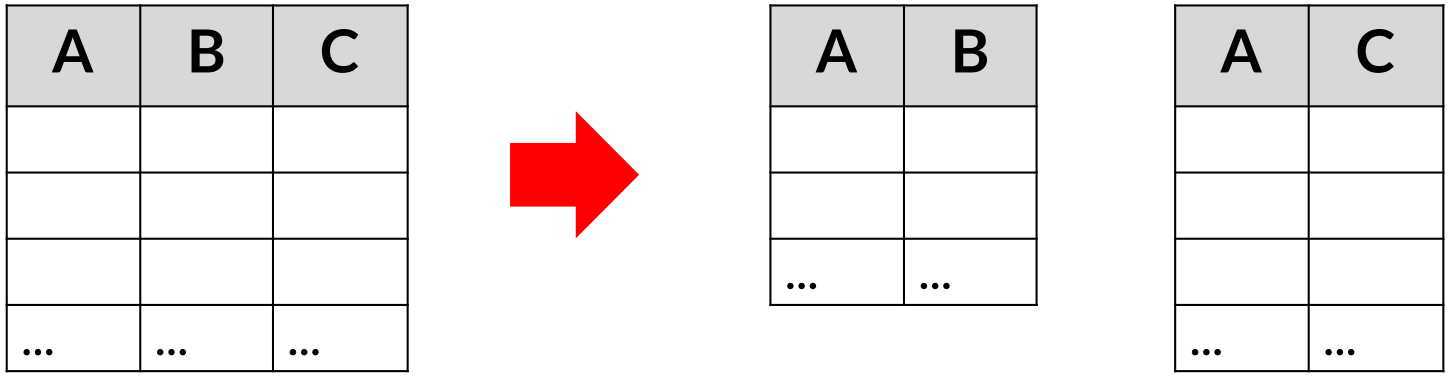
- Favorites(name, addr, favoriteBeer)
  - name $\longrightarrow$ addr
  - name $\longrightarrow$ favoriteBeer
- BCNF?

name	addr	favoriteBeer
Alex	100 Green St	Sam Adams
Carissa	200 Green St	Bud Light
Alex	100 Green St	Coors

Example Favorites table

BCNF Decomposition

- **Algorithm BCNF**
- Input: Relation R, FDs F
- If (exists an FD $A \longrightarrow B$ that violates the BCNF condition)
  - Decompose $R(A, B, C)$ into $R_1(A, B)$ and $R_2(A, C)$.
  - Compute FDs for R_1 and R_2 as F_1 and F_2.
  - Return BCNF(R_1, F_1) ∪ BCNF(R_2, F_2).
- Else
  - Return R.



BCNF decomposition

Decomposing into BCNF

- R = (id, name, major, birthday)
- F = {id ⟶ name, id ⟶ birthday}

id	name	major	birthday
1	Bugs Bunny	CS	2004-11-06
1	Bugs Bunny	Music	2004-11-06
2	Donald Duck	Bio	1997-02-01
3	Peter Pan	Econ	1998-10-01
3	Peter Pan	Social	1998-10-01
3	Peter Pan	ME	1998-10-01
4	Mickey Mouse	CS	1995-04-01

Example Students table

- FD f: id ⟶ name, birthday violates BCNF
- Decompose into:
  - R_1 = (id, name, birthday), F_1 = {id ⟶ name, id ⟶ birthday}
  - R_2 = (id, major), F_2 = ∅
- Done?

Decomposing into BCNF

- $R =$ (id, name, major, birthday, adviser)
- $F = \{$id $\longrightarrow$ name, id $\longrightarrow$ birthday, major $\longrightarrow$ adviser$\}$

- FD f: id $\longrightarrow$ name, birthday violates BCNF
- Decompose into:
  - $R_1 =$ (id, name, birthday), $F_1 = \{$id $\longrightarrow$ name, id $\longrightarrow$ birthday$\}$
  - $R_2 =$ (id, major, adviser), $F_2 = \{$major $\longrightarrow$ adviser$\}$
- Done?

Food for Thought

Is BCNF unique?
I.e., given a relation R and FDs F, does BCNF decomposition results in a unique BCNF?

- Students(id, name, phone)
- Suppose: id → name, name → id

- What BCNF exists?