Database design: E/R diagrams and BCNF

CSE 444 section

October 14, 2010

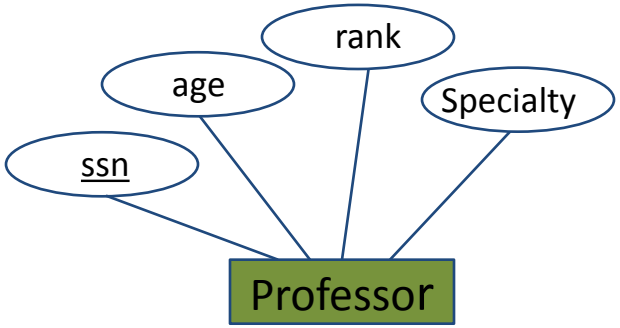
Today

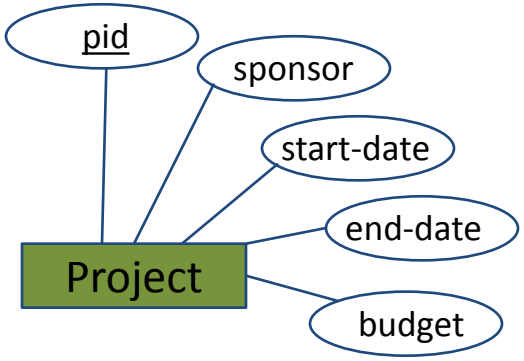
- Database design with E/R diagrams
- Functional dependencies
- Boyce-Codd normal form (BCNF)

Example from: R. Ramakrishnan and J. Gehrke, *Database Management Systems*, 3rd ed.

From English to an E/R diagram

- **Professors** have SSN, age, rank, and specialty
- **Projects** have IDs, sponsors, budgets, start and end dates

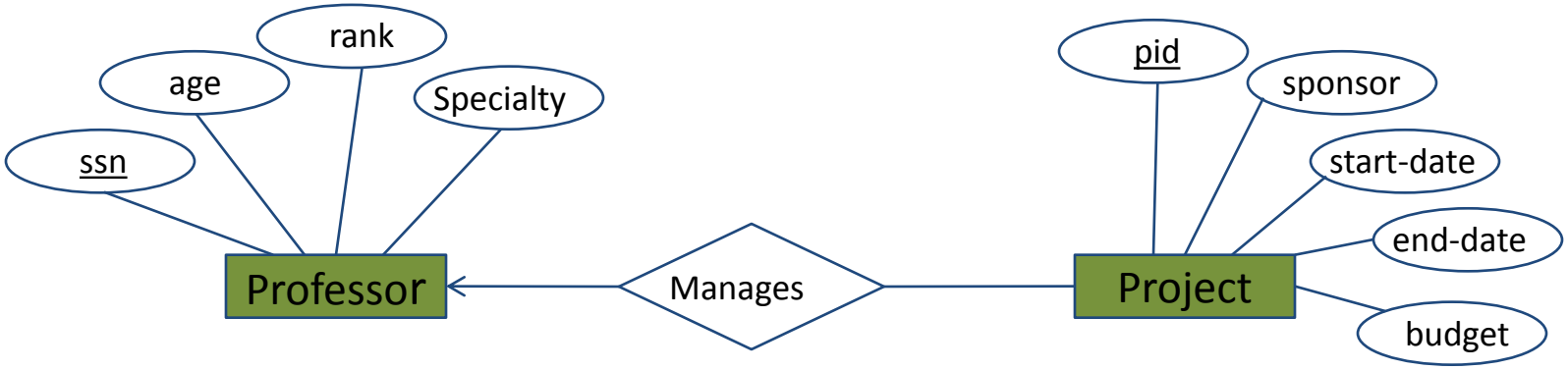




Example from: R. Ramakrishnan and J. Gehrke, *Database Management Systems*, 3rd ed.

From English to an E/R diagram

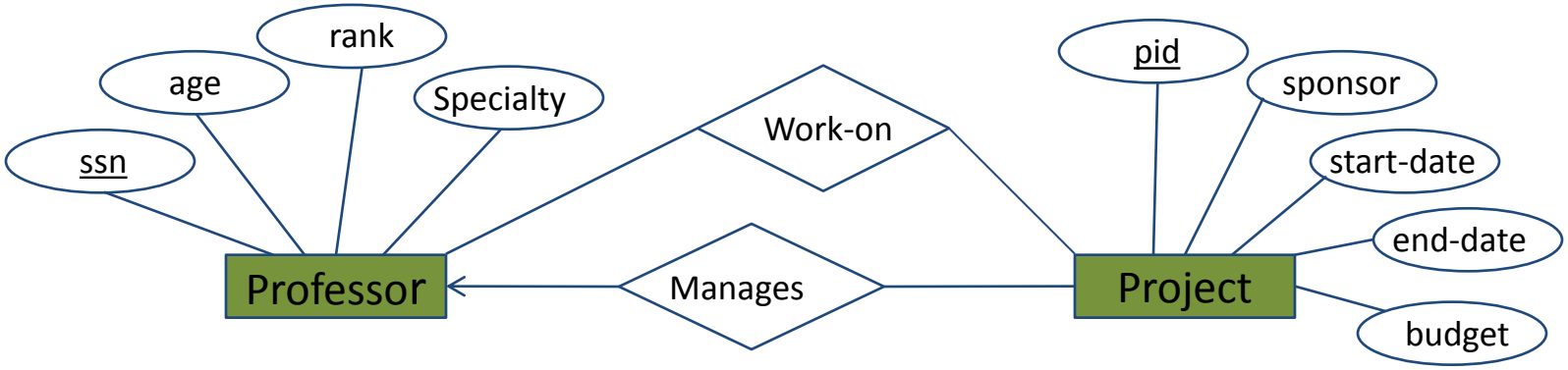
- Each project is **managed by** one professor (principal investigator)
- A professor can **manage** multiple projects



Example from: R. Ramakrishnan and J. Gehrke, *Database Management Systems*, 3rd ed.

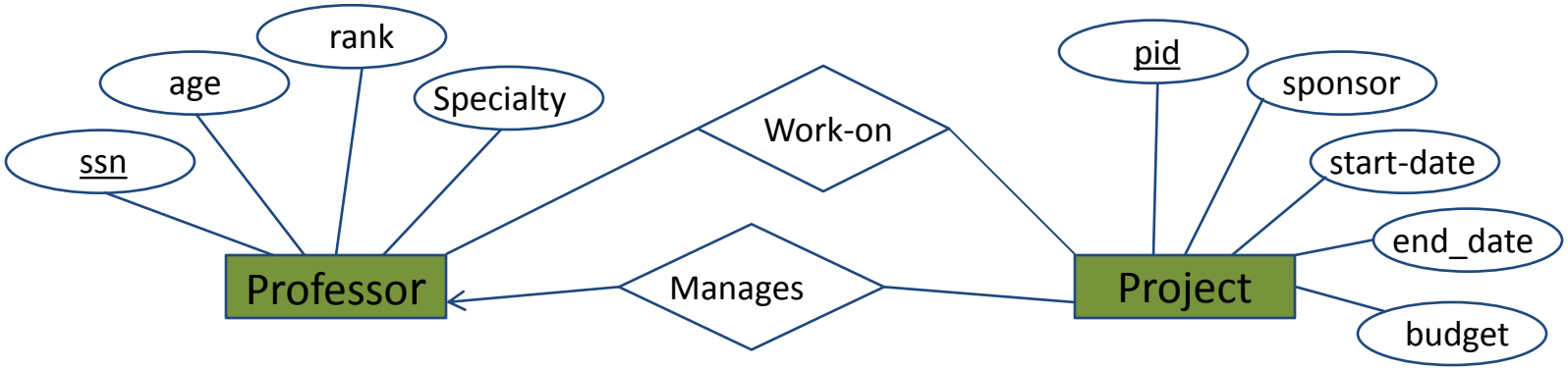
From English to an E/R diagram

- Each project is **worked on** by one or more professors
- Professors can **work on** multiple projects



Example from: R. Ramakrishnan and J. Gehrke, *Database Management Systems*, 3rd ed.

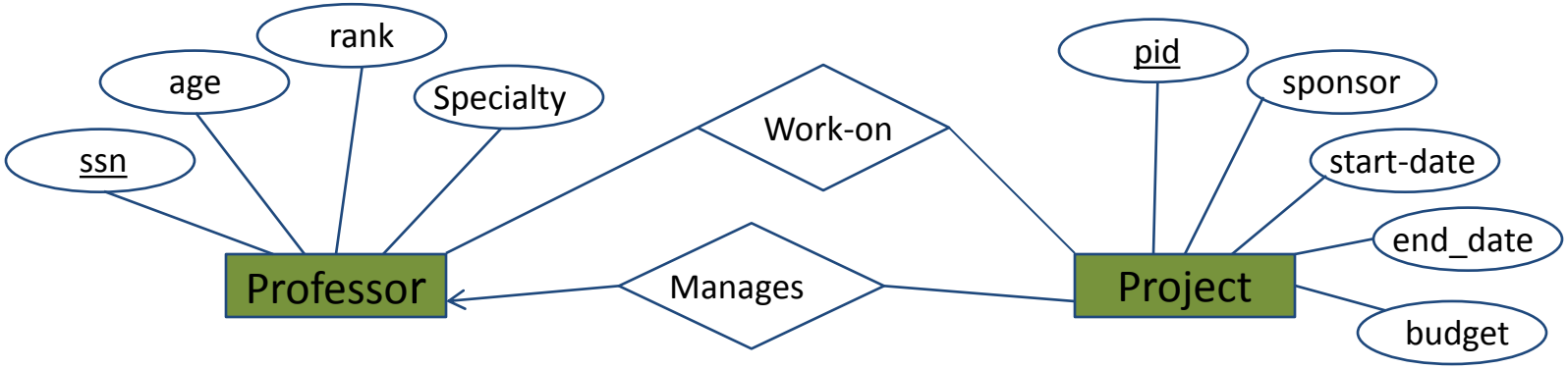
From E/R diagram to relations



- Professor (ssn, age, rank, specialty)
- Project (pid, sponsor, start_date, end_date, budget)
- Work_on (ssn, pid)
- Manages (ssn, pid)

Example from: R. Ramakrishnan and J. Gehrke, *Database Management Systems*, 3rd ed.

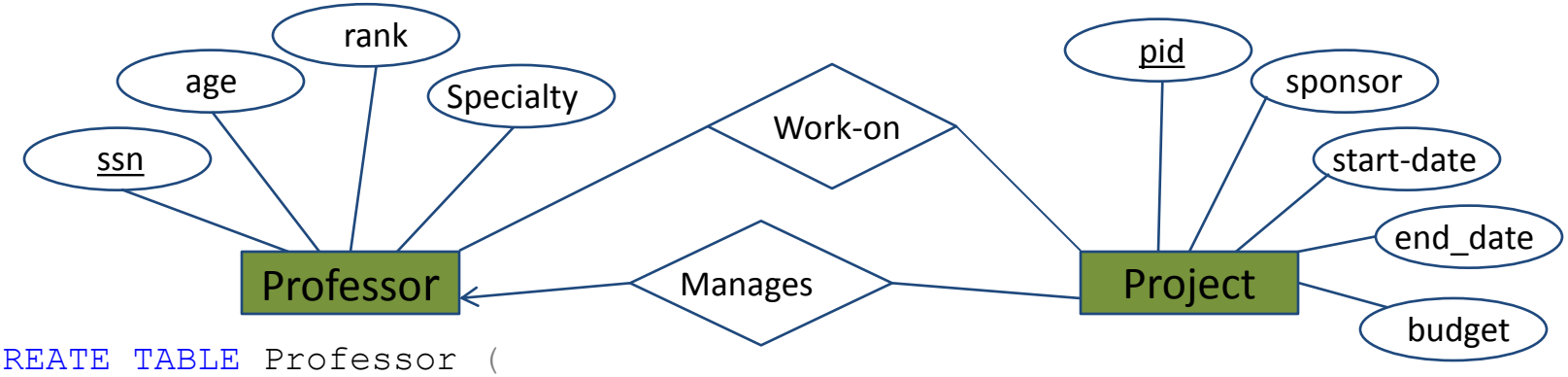
Integrating the many-one relation



- Professor (ssn, age, rank, specialty)
- Project (pid, sponsor, start_date, end_date, budget, ssn)
- Work_on (ssn, pid)

Example from: R. Ramakrishnan and J. Gehrke, *Database Management Systems*, 3rd ed.

SQL code for this database




```
CREATE TABLE Professor (
     ssn INT PRIMARY KEY,
     age INT,
     urank VARCHAR(30),
     specialty VARCHAR(30)
);
```

```
CREATE TABLE Work_on (
     ssn INT REFERENCES Professor(ssn),
     pid INT REFERENCES Project(pid),
     PRIMARY KEY (ssn, pid)
);
```

```
CREATE TABLE Project (
     pid INT PRIMARY KEY,
     sponser INT,
     start_date DATE,
     end_date DATE,
     budget FLOAT,
     ssn INT REFERENCES Professor(ssn)
);
```

- Professor(ssn, age, rank, specialty)
- Project(pid, sponsor, start_date, end_date, budget, ssn)
- Work_in(ssn, pid)

Today

- Database design with E/R diagrams
- Functional dependencies
- Boyce-Codd normal form (BCNF)

Garcia-Molina, problem 3.3.2 (i)

Consider a relation S(A,B,C,D) with FDs
A → B, B → C, and B → D.

a. Give the nontrivial FDs that follow from the given FDs. Restrict to 1 attr on right side.
b. What are all the keys of S?
c. What are the superkeys that aren't keys?

Garcia-Molina, problem 3.3.2 (ii)

Consider a relation T(A,B,C,D) with FDs
AB → C, BC → D, CD → A, and AD → B.

a. Give the nontrivial FDs that follow from the given FDs. Restrict to 1 attr on right side.
b. What are all the keys of S?
c. What are the superkeys that aren't keys?

Today

- Database design with E/R diagrams
- Functional dependencies
- Boyce-Codd normal form (BCNF)

What is BCNF?

A relation R is in BCNF iff:

If A_1,… $A_n \rightarrow B$ is a non-trivial dependency in R, then $\{A_1, \ldots, A_n\}$ is a superkey for R

Why do BCNF decompositions?

BCNF decomposition algorithm

BCNF_Decompose(R)

find X s.t.: X ≠X^+ ≠ [all attributes]

if (not found) **then** "R is in BCNF"

let Y = X^+ - X
let Z = [all attributes] - X^+
decompose R into R1(X ∪ Y) and R2(X ∪ Z)
continue to decompose recursively R1 and R2

BCNF example: table R(A, B, C, D, E)

Consider the following FDs:

- CD → E BAD
- D → B BAD
- A → CD

Which ones are the bad dependencies?

CD+ = BCDE

CD is not a superkey

D+ = BD

D is not a superkey

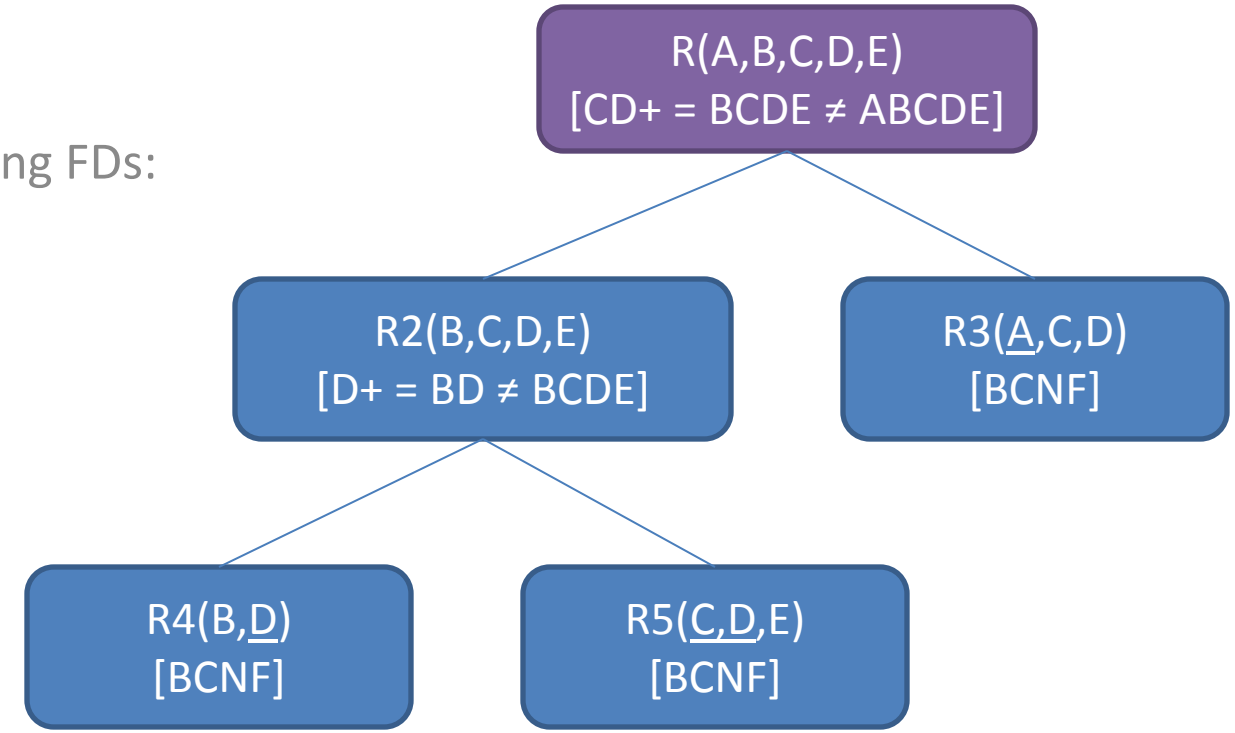
A+ = ABCDE

A is a superkey

BCNF example: table R(A, B, C, D, E)

Consider the following FDs:

- CD → E BAD
- D → B BAD
- A → CD



2 more BCNF decompositions

S(A, B, C, D)

C → D, C → A, B → C

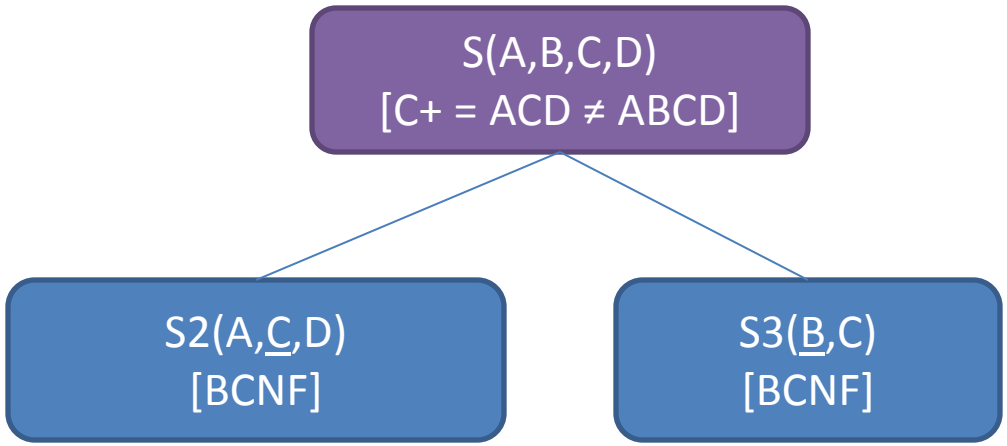
T(A, B, C, D, E)

AB → C, DE → C, B → D

S(A,B,C,D) solution

Consider the following FDs:

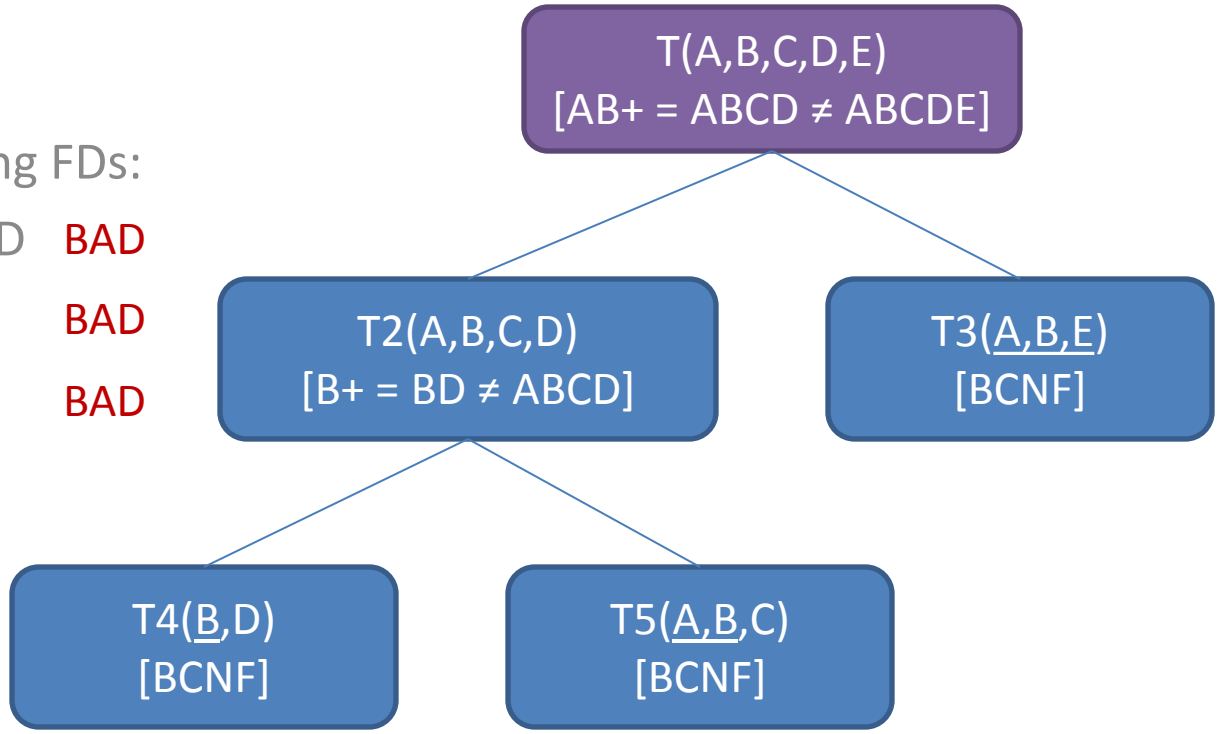
- C → D, C+ = ACD BAD
- C → A, C+ = ACD BAD
- B → C, B+ = ABCD



T(A,B,C,D,E) 1st solution

Consider the following FDs:

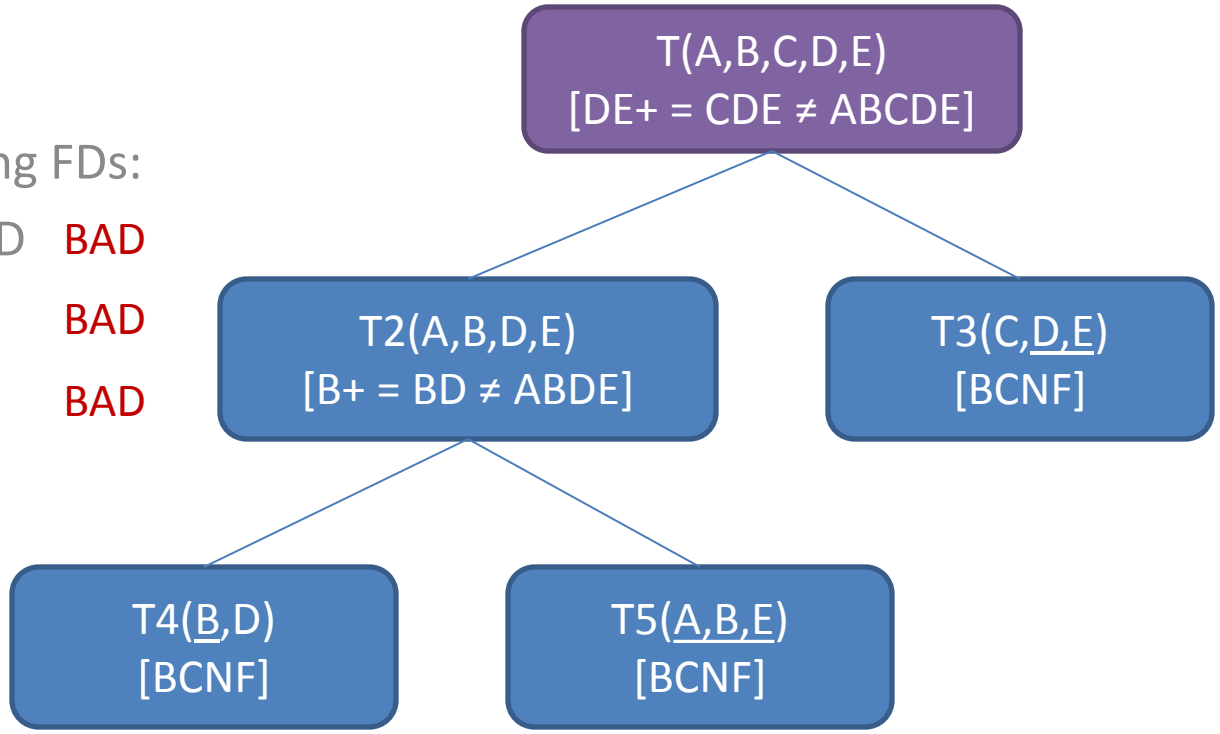
- AB → C, AB+ = ABCD BAD
- DE → C, DE+ = CDE BAD
- B → D, B+ = BD BAD



T(A,B,C,D,E) 2nd solution

Consider the following FDs:

- AB → C, AB+ = ABCD BAD
- DE → C, DE+ = CDE BAD
- B → D, B+ = BD BAD



T(A,B,C,D,E) 3rd solution

Consider the following FDs:

- AB → C, AB+ = ABCD BAD
- DE → C, DE+ = CDE BAD
- B → D, B+ = BD BAD