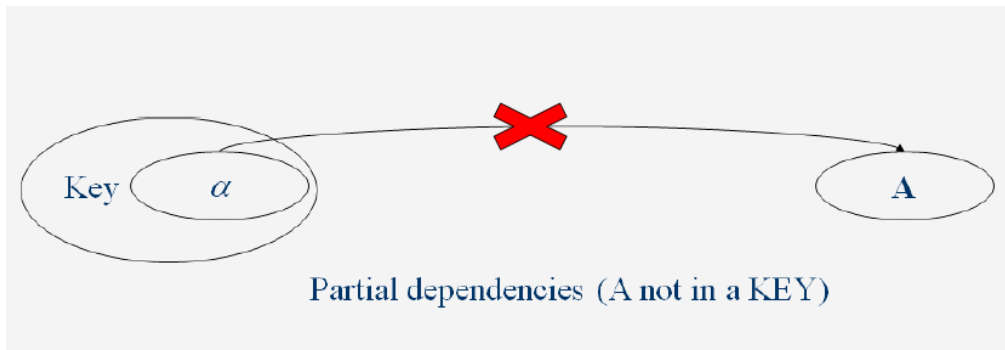


## FORME NORMALE

1. **1NF** – toate valorile atributelor sunt **atomice**
2. **2NF** – este **1NF** și toate attributele non-cheie depind de întreaga cheie (**nu are dependențe parțiale**)

Dependență funcțională parțială = un atribut **neprim** (care nu face parte dintr-o cheie) este dependent funcțional de o parte a cheii primare a relației (dar **nu** de întreaga cheie).

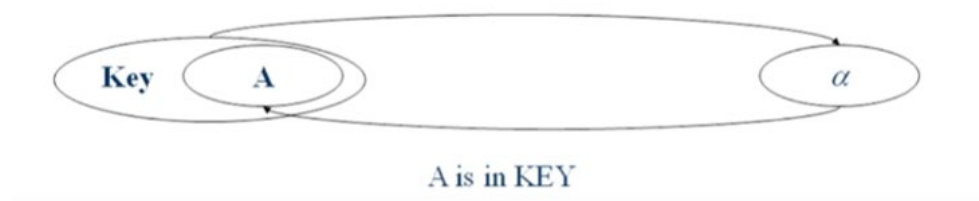


3. **BCNF** – este **2NF** și toate dependențele sunt date de chei

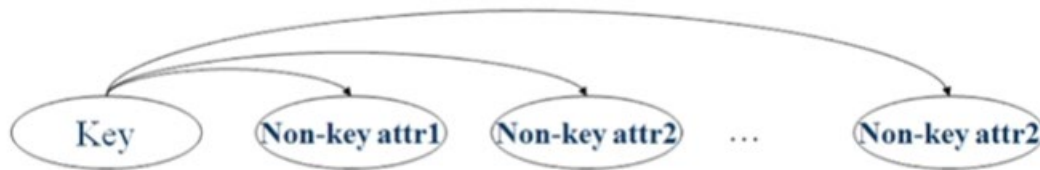
Cu alte cuvinte: R este în BCNF dacă singurele dependențe funcționale satisfăcute de R sunt cele corespunzătoare constrângerilor de cheie.

4. **3NF** – este **2NF** toate attributele non-primare depind doar de cheie (**nu sunt dependențe tranzitive**)

R este în BCNF => R este în 3NF



## BCNF & 3NF



Normalizarea pe scurt:

Fiecare atribut depinde:

de **cheie**, .....► definiție cheie

de **întreaga cheie**, .....► 2NF

și de nimic altceva

**decât de cheie** .....► BCNF

Fiecare atribut <sup>neprim</sup> depinde:

de **cheie**, .....► definiție cheie

de **întreaga cheie**, .....► 2NF

și de nimic altceva

**decât de cheie** .....► 3NF

### Descompunerea în BCNF:

Fie relația R cu dependențele funcționale F. Dacă  $\alpha \rightarrow A$  nu respectă BCNF ( $\alpha$  nu este cheie), atunci descompunem R în  $R - A$  și  $\alpha A$ . Se aplică repetat.

EX:

$R(\underline{C}, S, J, D, P, Q, V)$ ,  $F = \{JP \rightarrow C, SD \rightarrow P, J \rightarrow S\}$  dep. funcționale

Niciuna dintre dependențe nu respectă BCNF (în stânga nu se află cheia C).

- Alegem  $SD \rightarrow P$  și descompunem în  $(\underline{C}, S, J, D, Q, V)$  (acel  $R - A$ ) și  $(\underline{S}, \underline{D}, P)$  (acel  $\alpha A$ ).
- Alegem  $J \rightarrow S$  și descompunem în  $(\underline{C}, J, D, Q, V)$  și  $(\underline{J}, S)$ .

### Descompunerea în 3NF:

Folosim procedeul descompunerii în BCNF, dar, în loc de a utiliza mulțimea dependențelor funcționale F, folosim o **acoperire minimală a lui F**. În plus, dependențele funcționale trebuie să respecte 3NF.

Algoritmul de calcul al acoperirii minimele pentru F:

1. Folosim descompunerea pentru a obține dependențe funcționale cu un atribut în partea dreaptă.
2. Se elimină attributele redundante.
3. Se elimină dependențele funcționale redundante.

EX: Fie  $F = \{ABCD \rightarrow E, E \rightarrow D, A \rightarrow B, AC \rightarrow D\}$ .

Atributele BD din  $ABCD \rightarrow E$  sunt redundante (pentru că  $A \rightarrow B$  și  $AC \rightarrow D$ , deci valorile B și D pot fi obținute pe baza valorilor A și C).

$\Rightarrow F = \{AC \rightarrow E, E \rightarrow D, A \rightarrow B, AC \rightarrow D\}$ .

$AC \rightarrow D$  este redundantă (deoarece din dep.  $AC \rightarrow E$  și  $E \rightarrow D$  reiese că  $AC \rightarrow D$ ).

$\Rightarrow F = \{AC \rightarrow E, E \rightarrow D, A \rightarrow B\}$ , care este o acoperire minimală.

EX descompunere în 3NF:

Fie  $R(A, B, C, D, E)$ ,  $F = \{ABCD \rightarrow E, E \rightarrow D, A \rightarrow B, AC \rightarrow D\}$  dep. funcționale.

- Acoperirea minimală a lui F este  $\{AC \rightarrow E, E \rightarrow D, A \rightarrow B\}$ .
  - Unica cheie: AC (pentru că știind A și C putem obține toate celelalte atribute)
  - R nu e în 3NF deoarece  $A \rightarrow B$  nu respectă 3NF
  - Descompunerea:
    - Relații pt. fiecare DF:  $R_1(A, C, E)$ ,  $R_2(E, D)$ ,  $R_3(A, B)$
    - Relație pt. cheia lui R:  $R_4(A, C)$ , dar o eliminăm pt. că e redundantă
- $\Rightarrow$  descompunerea 3NF este  $\{R_1(A, C, E), R_2(E, D), R_3(A, B)\}$