COSC 3380
Design of Database Systems

Functional Dependencies and Normalization for Relational Databases

April 3, 2024

Second Normal Form

- A relation schema R is in **second normal form (2NF)** if every non-prime attribute A in R is fully functionally dependent on any key of R.

- Test for 2NF: Check FDs whose left-hand side attributes are part of the primary key
  - If the primary key contains a single attribute, no test needed.

Third Normal Form

- ***Transitive dependency***
  - X→Y in a relation R is a transitive dependency, if
    - there exists a set of attributes Z in R which is neither a candidate key or a subset of any key
    - both X→Z and Z→Y hold

Third Normal Form

Definition. A relation schema R is in **third normal form** (3NF) if, whenever a *nontrivial* functional dependency $X \rightarrow A$ holds in R, either (a) X is a superkey of R, or (b) A is a prime attribute of R.

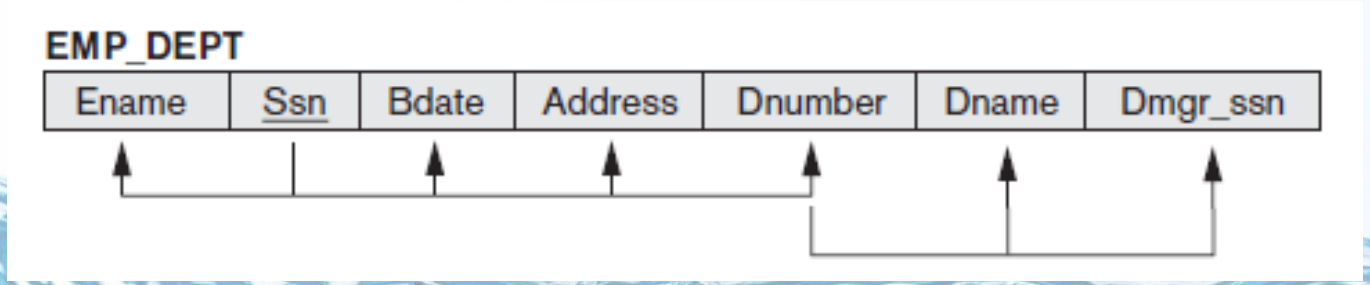
Alternative Definition. A relation schema R is in 3NF if every nonprime attribute of R meets both of the following conditions:

- It is fully functionally dependent on every key of R.
- It is nontransitively dependent on every key of R.

Third Normal Form

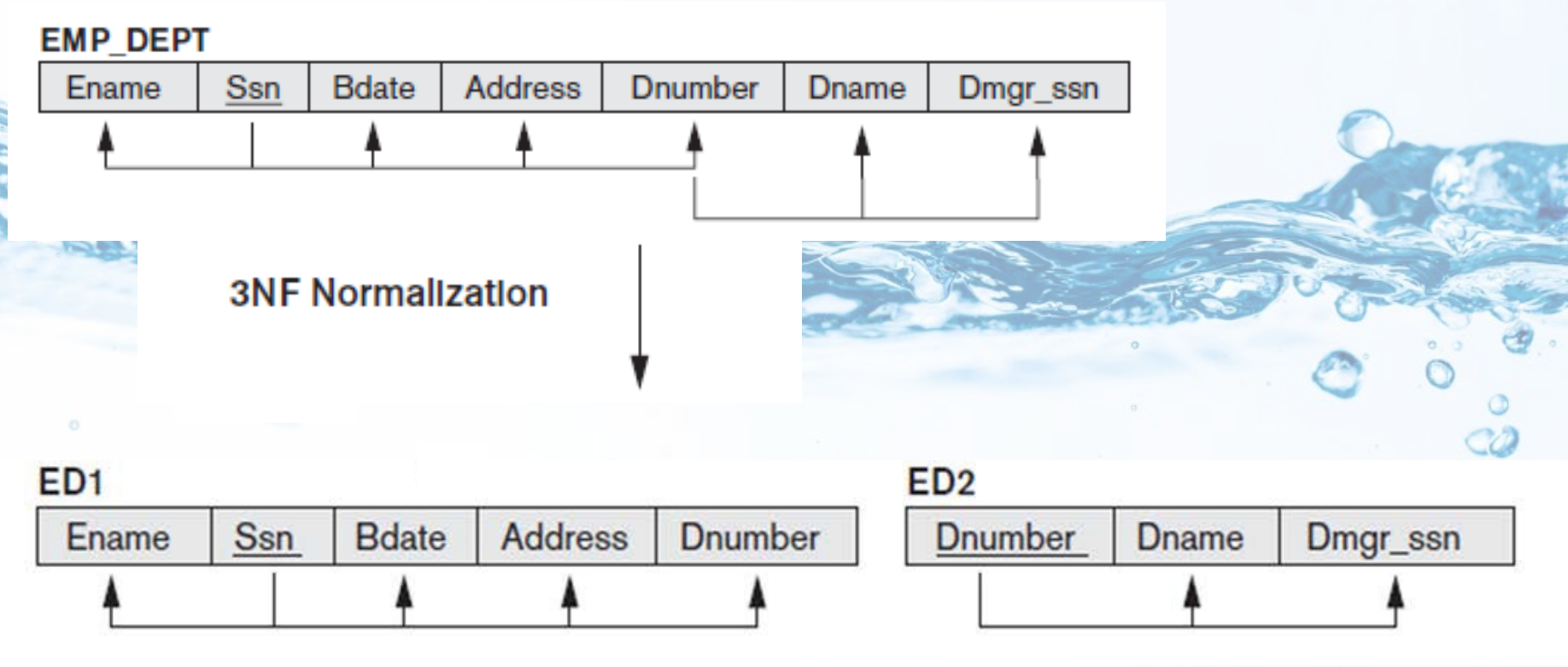
- A relation schema is in 3NF –
  - if it satisfies 2NF
  - no nonprime attribute of R is transitively dependent on the primary key.

Third Normal Form

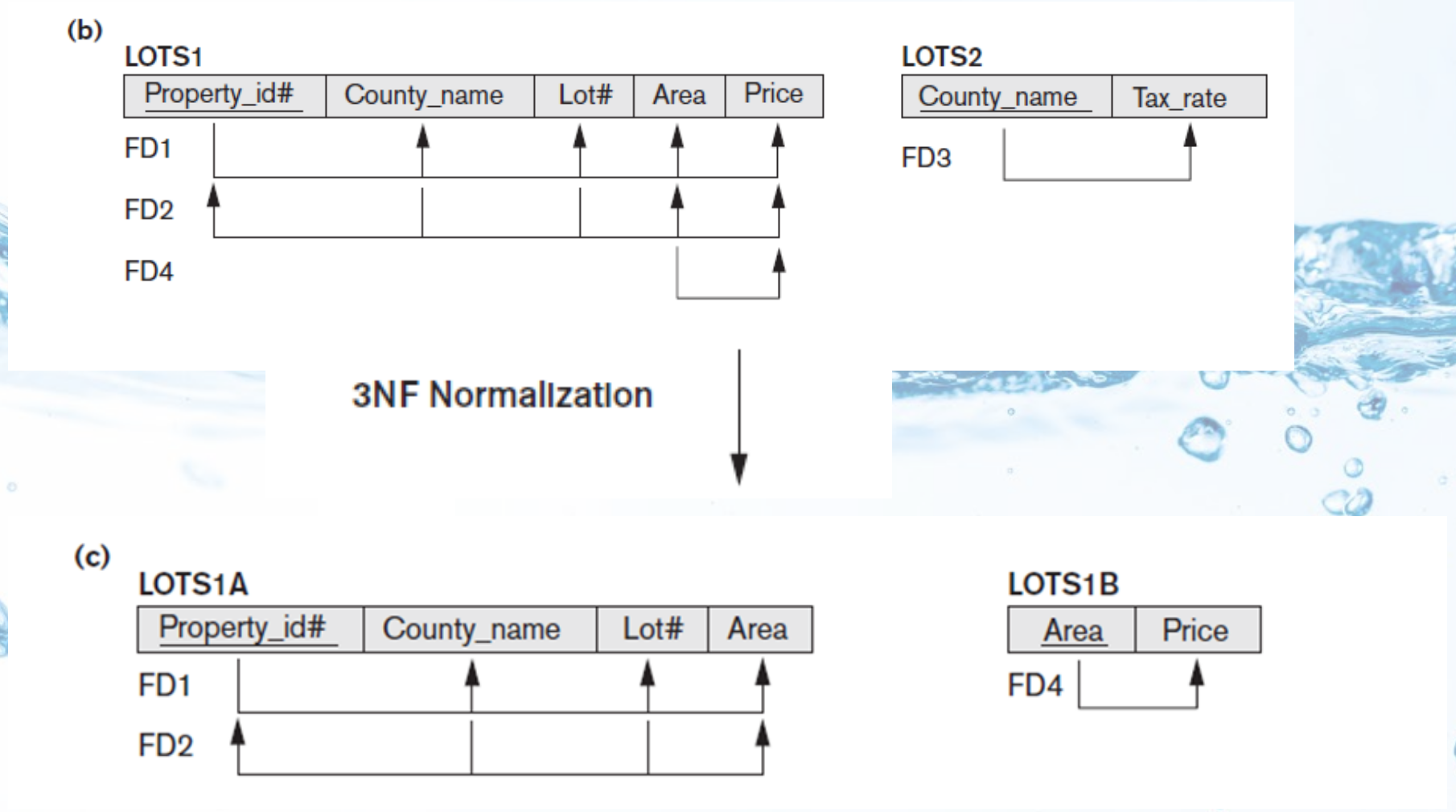


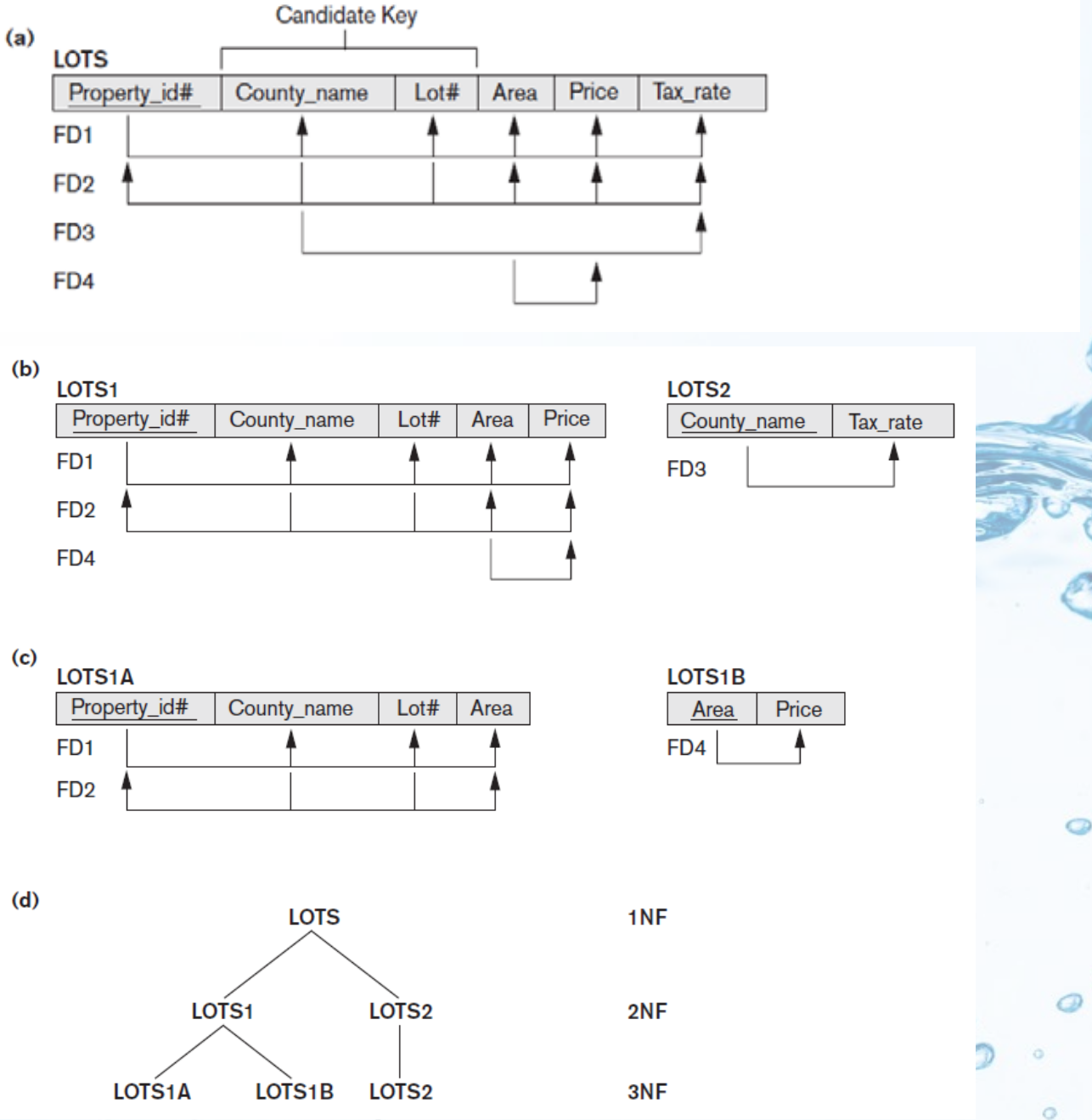
- Dependency **Ssn → Dmgr_ssn** is transitive through **Dnumber** in EMP_DEPT
- **Ssn → Dnumber** and **Dnumber → Dmgr_ssn** hold
- **Dnumber** is neither a key itself; nor a subset of the key of EMP_DEPT

Third Normal Form



Third Normal Form




(a)
Candidate Key
LOTS
Property_id#
County_name
Lot#
Area
Price
Tax_rate
FD1
FD2
FD3
FD4
(b)
LOTS1
Property_id#
County_name
Lot#
Area
Price
FD1
FD2
FD4
LOTS2
County_name
Tax_rate
FD3
(c)
LOTS1A
Property_id#
County_name
Lot#
Area
FD1
FD2
LOTS1B
Area
Price
FD4
(d)
LOTS
LOTS1
LOTS2
LOTS1A
LOTS1B
LOTS2
1NF
2NF
3NF

Normal Forms Defined Informally

- First normal form (1NF)
  - All attributes depend on **the key**
- Second normal form (2NF)
  - All attributes depend on **the whole key**
- Third normal form (3NF)
  - All attributes depend on **nothing but the key**

First, Second and Third Normal Forms

Normal Form	Test	Remedy (Normalization)
First (1NF)	Relation should have no multivalued attributes or nested relations.	Form new relations for each multivalued attribute or nested relation.
Second (2NF)	For relations where primary key contains multiple attributes, no nonkey attribute should be functionally dependent on a part of the primary key.	Decompose and set up a new relation for each partial key with its dependent attribute(s). Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it.
Third (3NF)	Relation should not have a nonkey attribute functionally determined by another nonkey attribute (or by a set of nonkey attributes). That is, there should be no transitive dependency of a nonkey attribute on the primary key.	Decompose and set up a relation that includes the nonkey attribute(s) that functionally determine(s) other nonkey attribute(s).

BCNF (Boyce-Codd Normal Form)

Definition. A relation schema R is in **third normal form** (3NF) if, whenever a *nontrivial* functional dependency $X \rightarrow A$ holds in R, either (a) X is a superkey of R, or (b) A is a prime attribute of R.

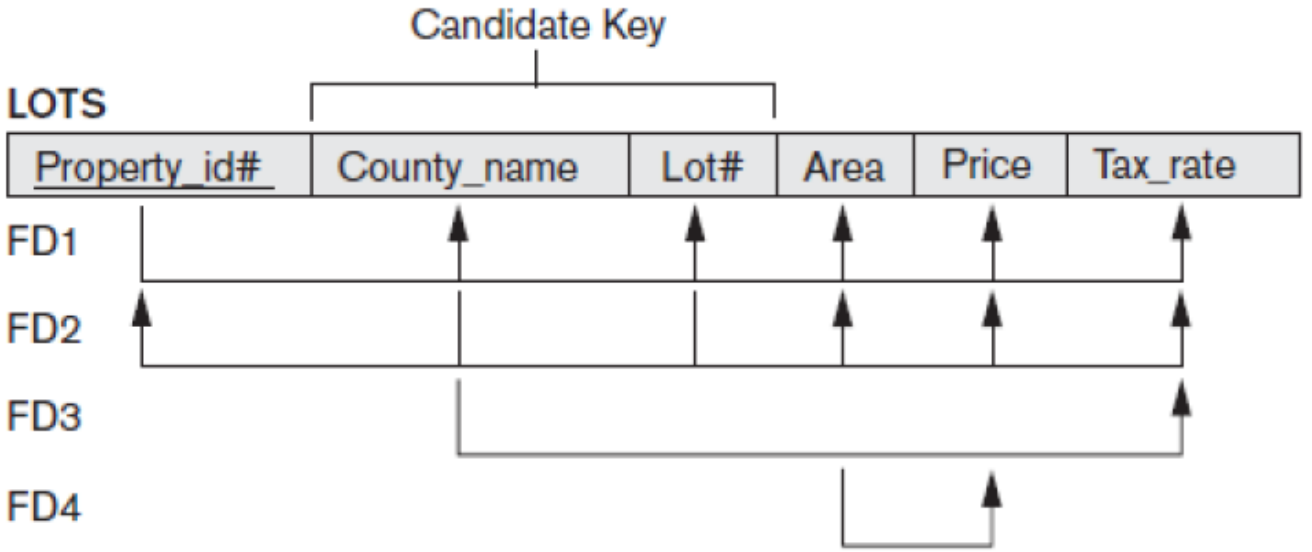
Definition. A relation schema R is in **BCNF** if whenever a *nontrivial* functional dependency $X \rightarrow A$ holds in R, then X is a superkey of R.

- Difference from 3NF
  - FDs having the RHS as a prime attribute [clause (b)] is absent

BCNF (Boyce-Codd Normal Form)

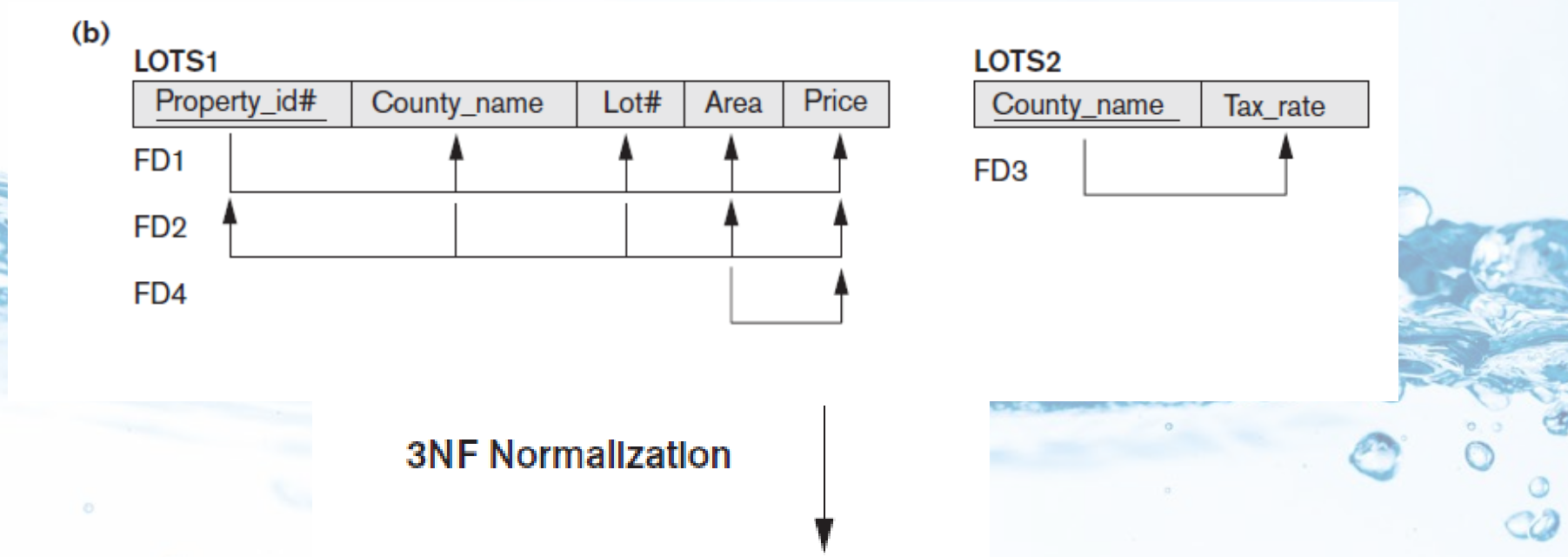
- A relation schema R is in **Boyce-Codd Normal Form (BCNF)** if whenever a **FD X → A** holds in R, then **X is a superkey** of R
- Each normal form is strictly stronger than the previous one
  - Every 2NF relation is in 1NF
  - Every 3NF relation is in 2NF
  - Every BCNF relation is in 3NF
- Relations can be in 3NF, but not in BCNF
- BCNF is considered a stronger form of 3NF
- The goal is to have each relation in BCNF (or 3NF)

BCNF (Boyce-Codd Normal Form)

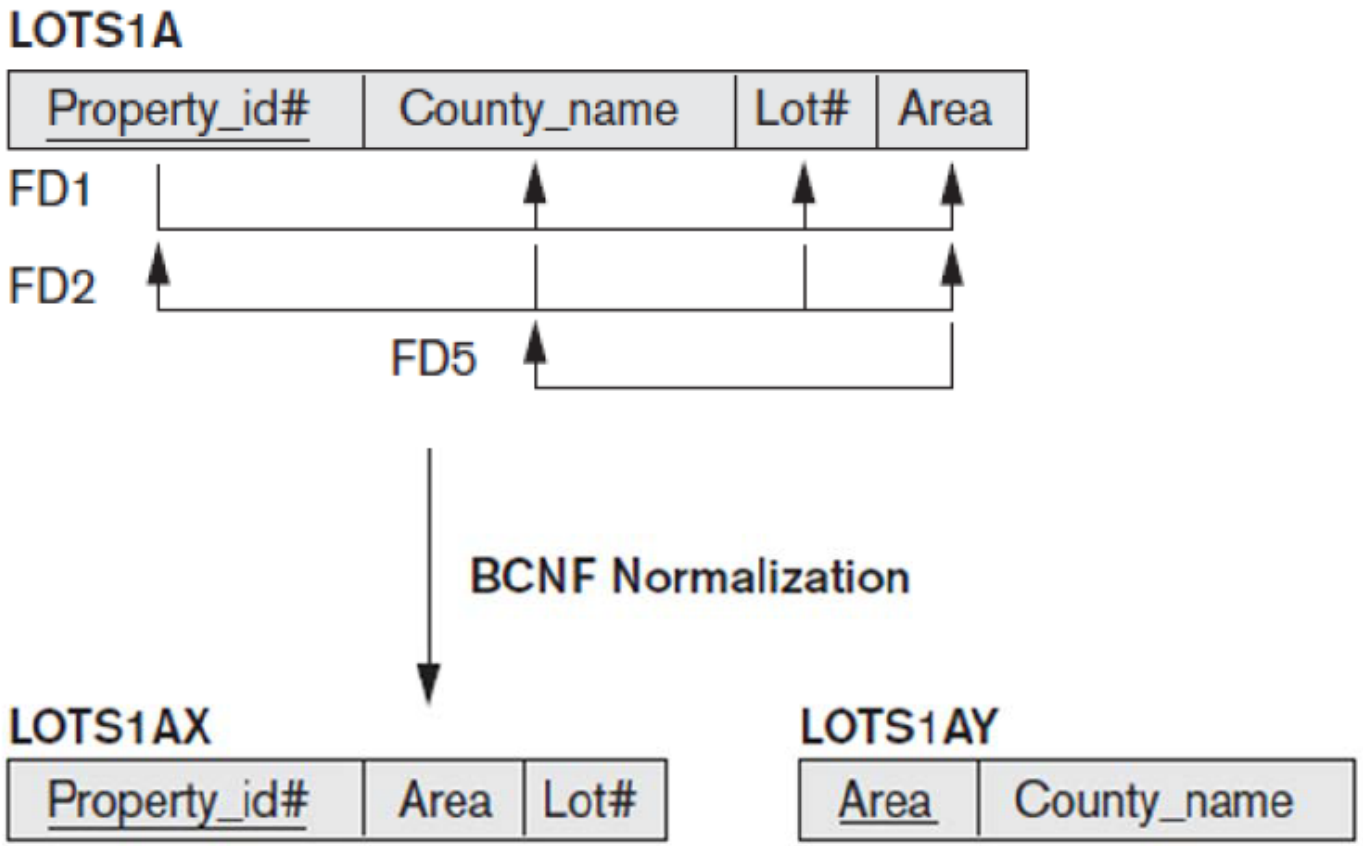


- Thousands of lots
- Lots from only two counties
- Lot sizes are different in the 2 counties
  - additional dependency
  - FD5: Area → County_name

BCNF (Boyce-Codd Normal Form)

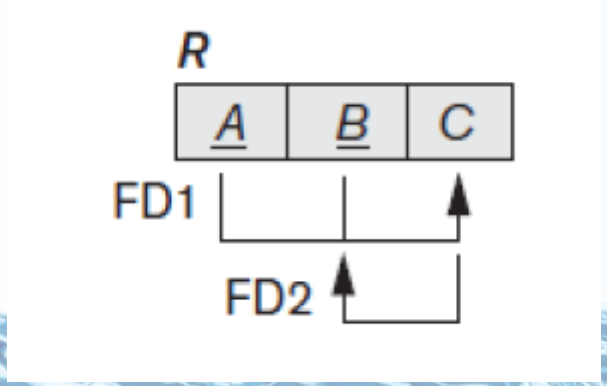


BCNF (Boyce-Codd Normal Form)



FD2 is lost in this decomposition

BCNF (Boyce-Codd Normal Form)

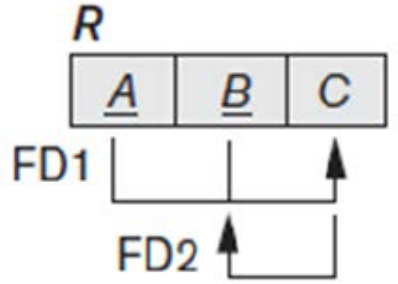


- R is in 3NF, but not BCNF due to FD2
- Such a FD results in potential redundancy of data
- Ideally you want to achieve BCNF or 3NF for every relation

BCNF (Boyce-Codd Normal Form)

TEACH

Student	Course	Instructor
Narayan	Database	Mark
Smith	Database	Navathe
Smith	Operating Systems	Ammar
Smith	Theory	Schulman
Wallace	Database	Mark
Wallace	Operating Systems	Ahamad
Wong	Database	Omiecinski
Zelaya	Database	Navathe
Narayan	Operating Systems	Ammar



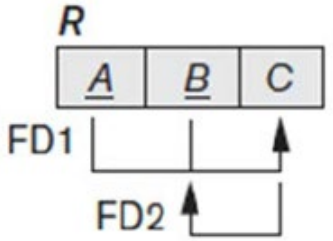
{student, course} is a candidate key

FD1: {Student, Course} → Instructor
FD2: Instructor → Course

BCNF (Boyce-Codd Normal Form)

Possible decompositions of TEACH(Student, Course, Instructor):

1. {Student, Instructor} and {Student, Course}
2. {Course, Instructor} and {Course, Student}
3. {Instructor, Course} and {Instructor, Student}



FD1 is lost in these decompositions

3NF and BCNF (Boyce-Codd Normal Form)

BASIS FOR COMPARISON	3NF	BCNF
Concept	No non-prime attribute must be transitively dependent on the Candidate key.	For any trivial dependency in a relation R say X->Y, X should be a super key of relation R.
Dependency	3NF can be obtained without sacrificing all dependencies.	Dependencies may not be preserved in BCNF.
Decomposition	Lossless decomposition can be achieved in 3NF.	Lossless decomposition is hard to achieve in BCNF.

Multivalued Dependency and 4NF

- Multivalued dependency (MVD)
  - Consequence of first normal form (1NF)
  - Two or more multivalued independent attributes in the same relation
  - Repeat every value of one of the attributes with every other value of other attribute to maintain data consistency and data independence

Multivalued Dependency and 4NF

EMP

Ename	Pname	Dname
Smith	X	John
Smith	Y	Anna
Smith	X	Anna
Smith	Y	John

EMP_PROJECTS

Ename	Pname
Smith	X
Smith	Y

EMP_DEPENDENTS

Ename	Dname
Smith	John
Smith	Anna

Ename $\twoheadrightarrow$ Pname

Ename $\twoheadrightarrow$ Dname

When two independent 1:N relationships are mixed in the same Relation, an MVD may occur.

Fourth Normal Form

Definition. A relation schema R is in 4NF with respect to a set of dependencies F (that includes functional dependencies and multivalued dependencies) if, for every *nontrivial* multivalued dependency $X \twoheadrightarrow Y$ in F, X is a superkey for R.

Restaurant	Pizza Variety	Delivery Area
A1 Pizza	Thick Crust	Springfield
A1 Pizza	Thick Crust	Shelbyville
A1 Pizza	Thick Crust	Capital City
A1 Pizza	Stuffed Crust	Springfield
A1 Pizza	Stuffed Crust	Shelbyville
A1 Pizza	Stuffed Crust	Capital City
Elite Pizza	Thin Crust	Capital City
Elite Pizza	Stuffed Crust	Capital City
Vincenzo's Pizza	Thick Crust	Springfield
Vincenzo's Pizza	Thick Crust	Shelbyville
Vincenzo's Pizza	Thin Crust	Springfield
Vincenzo's Pizza	Thin Crust	Shelbyville

Varieties By Restaurant

Restaurant	Pizza Variety
A1 Pizza	Thick Crust
A1 Pizza	Stuffed Crust
Elite Pizza	Thin Crust
Elite Pizza	Stuffed Crust
Vincenzo's Pizza	Thick Crust
Vincenzo's Pizza	Thin Crust

Delivery Areas By Restaurant

Restaurant	Delivery Area
A1 Pizza	Springfield
A1 Pizza	Shelbyville
A1 Pizza	Capital City
Elite Pizza	Capital City
Vincenzo's Pizza	Springfield
Vincenzo's Pizza	Shelbyville

Join Dependencies and Fifth Normal Form

- Join dependency
  - Whenever a supplier ***s*** supplies part ***p***, and a project ***j*** uses part ***p***, and the supplier ***s*** supplies at least one part to project ***j***, then supplier ***s*** will also be supplying part ***p*** to project ***j***.
- Peculiar semantic constraint
  - Normalization into 5NF is very rarely done in practice.

Join Dependencies and Fifth Normal Form

SUPPLY

Sname	Part_name	Proj_name
Smith	Bolt	ProjX
Smith	Nut	ProjY
Adamsky	Bolt	ProjY
Walton	Nut	ProjZ
Adamsky	Nail	ProjX
Adamsky	Bolt	ProjX
Smith	Bolt	ProjY

Multiway decomposition

R_1

Sname	Part_name
Smith	Bolt
Smith	Nut
Adamsky	Bolt
Walton	Nut
Adamsky	Nail

R_2

Sname	Proj_name
Smith	ProjX
Smith	ProjY
Adamsky	ProjY
Walton	ProjZ
Adamsky	ProjX

R_3

Part_name	Proj_name
Bolt	ProjX
Nut	ProjY
Bolt	ProjY
Nut	ProjZ
Nail	ProjX

First, Second and Third Normal Forms

Normal Form	Test	Remedy (Normalization)
First (1NF)	Relation should have no multivalued attributes or nested relations.	Form new relations for each multivalued attribute or nested relation.
Second (2NF)	For relations where primary key contains multiple attributes, no nonkey attribute should be functionally dependent on a part of the primary key.	Decompose and set up a new relation for each partial key with its dependent attribute(s). Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it.
Third (3NF)	Relation should not have a nonkey attribute functionally determined by another nonkey attribute (or by a set of nonkey attributes). That is, there should be no transitive dependency of a nonkey attribute on the primary key.	Decompose and set up a relation that includes the nonkey attribute(s) that functionally determine(s) other nonkey attribute(s).