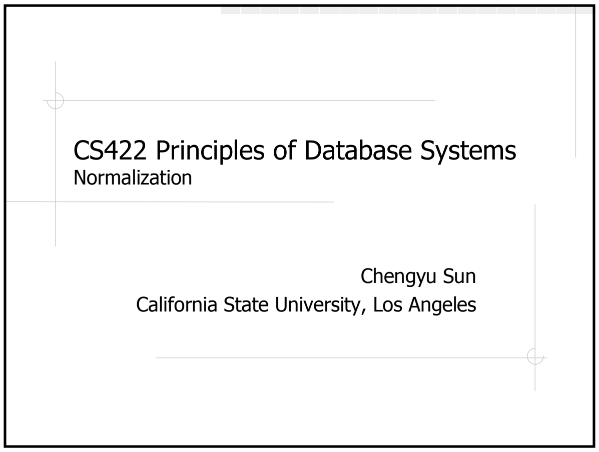
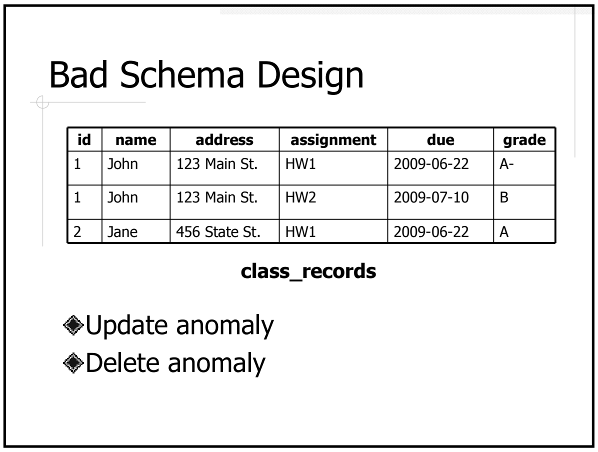


CS422 Principles of Database Systems

Normalization

Chengyu Sun
California State University, Los Angeles

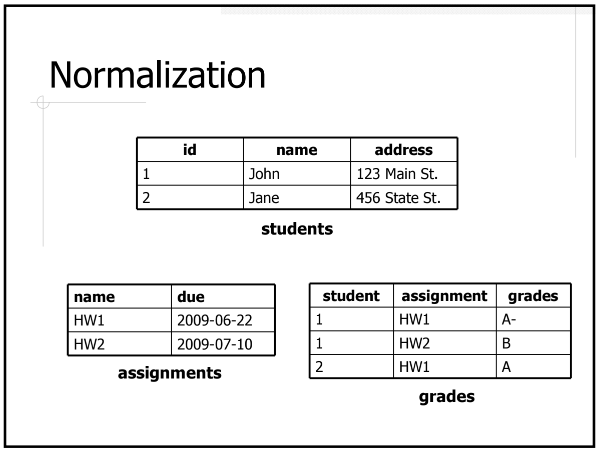


Bad Schema Design

id	name	address	assignment	due	grade
1	John	123 Main St.	HW1	2009-06-22	A-
1	John	123 Main St.	HW2	2009-07-10	B
2	Jane	456 State St.	HW1	2009-06-22	A

class_records

- Update anomaly
- Delete anomaly



Normalization

id	name	address
1	John	123 Main St.
2	Jane	456 State St.

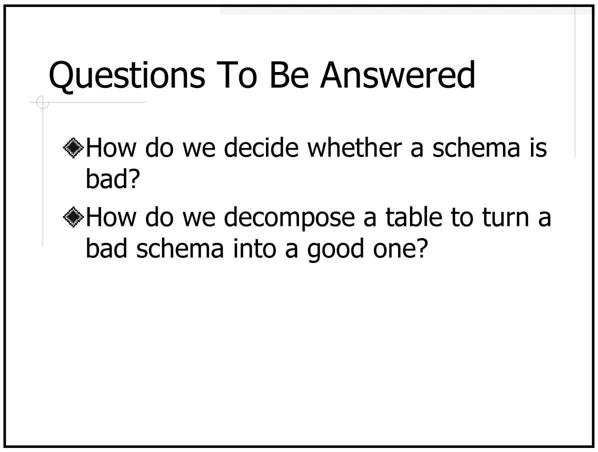
students

name	due
HW1	2009-06-22
HW2	2009-07-10

assignments

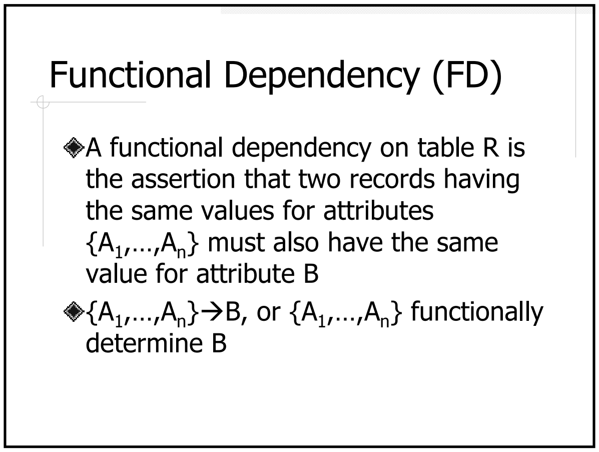
student	assignment	grades
1	HW1	A-
1	HW2	B
2	HW1	A

grades



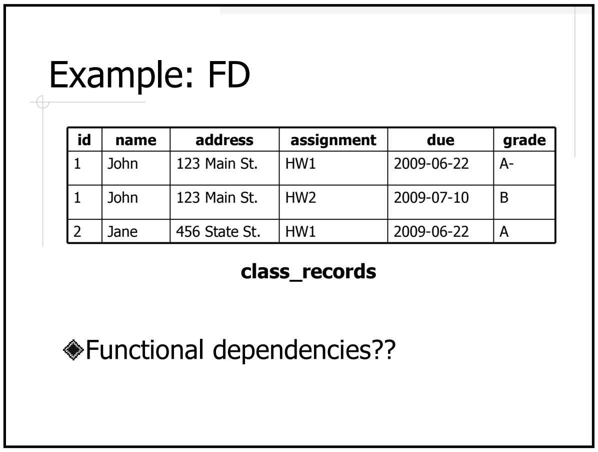
Questions To Be Answered

- How do we decide whether a schema is bad?
- How do we decompose a table to turn a bad schema into a good one?



Functional Dependency (FD)

- A functional dependency on table R is the assertion that two records having the same values for attributes $\{A_1,\ldots,A_n\}$ must also have the same value for attribute B
- $\{A_1,\ldots,A_n\} \rightarrow B$, or $\{A_1,\ldots,A_n\}$ functionally determine B



Example: FD

id	name	address	assignment	due	grade
1	John	123 Main St.	HW1	2009-06-22	A-
1	John	123 Main St.	HW2	2009-07-10	B
2	Jane	456 State St.	HW1	2009-06-22	A

class_records

- Functional dependencies??

FD with Multiple Attributes

$$\{A_1, A_2, A_3, \ldots, A_n\} \rightarrow B_1$$
$$\{A_1, A_2, A_3, \ldots, A_n\} \rightarrow B_2$$
$$\ldots$$
$$\{A_1, A_2, A_3, \ldots, A_n\} \rightarrow B_m$$

⇕

$$\{A_1, A_2, A_3, \ldots, A_n\} \rightarrow \{B_1, B_2, B_3, \ldots, B_m\}$$

⇕

$$\mathbf{A} \rightarrow \mathbf{B}$$

Trivial Functional Dependency

FD: $\{A_1, A_2, A_3, \ldots, A_n\} \rightarrow \{B_1, B_2, B_3, \ldots, B_m\}$

- FD is trivial if all B's are in **A**
- FD is nontrivial if at least one B is not in **A**
- FD is completely nontrivial if no B is in **A**

Key

- **A** is a key of table R if
  - **A** functionally determines all attributes of R
  - No proper subset of **A** functionally determines all attributes of R

A Few Things about Keys

- A table may have multiple keys
- A key may consist of multiple attributes
- Superset of a key is called a super key
- A key has to be *minimal*, but not necessarily *minimum*
- The definition doesn't say anything about *uniqueness*

Example: Key

id	name	address	assignment	due	grade
1	John	123 Main St.	HW1	2009-06-22	A-
1	John	123 Main St.	HW2	2009-07-10	B
2	Jane	456 State St.	HW1	2009-06-22	A

class_records

- Key??

Boyce-Codd Normal Form (BCNF)

- A table R is in BCNF if for every *nontrivial FD* **A** → B in R, **A** is a *super key* of R**.**

Or

The key, the whole key, and nothing but the key, so help me Codd.

Decompose into BCNF

- Given table **R** with FD's F
- Look among F for a BCNF violation **A**→B
- Compute $\mathbf{A}^+$
- Decompose R into:
  - $\mathbf{R_1} = \mathbf{A}^+$
  - $\mathbf{R_2} = (\mathbf{R} - \mathbf{A}^+) \cup \mathbf{A}$
- *Continue decomposition with* $\boldsymbol{R_1}$ *and* $\boldsymbol{R_2}$ until all resulting tables are BCNF

Closure of Attributes $\mathbf{A}^+$

- Given
  - a set of attributes **A**
  - a set of functional dependencies **S**
- Closure of **A** under **S**, $\mathbf{A}^+$, is the set of all possible attributes that are functionally determined by **A** based on the functional dependencies inferable from **S**

Simple Closure Example

- R: {A,B,C}
  - S: {A→B, B→C}
- $\{A\}^+$??
- $\{B\}^+$??
- $\{C\}^+$??

Armstrong's Axioms

Reflexivity

If $\mathbf{B} \subseteq \mathbf{A}$, then $\mathbf{A} \rightarrow \mathbf{B}$

Transitivity

If $\mathbf{A} \rightarrow \mathbf{B}$ and $\mathbf{B} \rightarrow \mathbf{C}$, then $\mathbf{A} \rightarrow \mathbf{C}$

Augmentation

If $\mathbf{A} \rightarrow \mathbf{B}$, then $\mathbf{AC} \rightarrow \mathbf{BC}$ for any **C**

Two More FD Rules

Union

If $\mathbf{A} \rightarrow \mathbf{B}$ and $\mathbf{A} \rightarrow \mathbf{C}$, then $\mathbf{A} \rightarrow \mathbf{BC}$

Decomposition

If $\mathbf{A} \rightarrow \mathbf{BC}$, then $\mathbf{A} \rightarrow \mathbf{B}$ and $\mathbf{A} \rightarrow \mathbf{C}$

Computing $\mathbf{A}^+$

- Initialize $\mathbf{A}^+ = \mathbf{A}$
- Search in **S** for **B**→C where
  - $\mathbf{B} \subseteq \mathbf{A}^+$
  - $C \notin \mathbf{A}^+$
- Add C to $\mathbf{A}^+$
- Repeat until nothing can be added to $\mathbf{A}^+$

Computing $\mathbf{A}^+$ Example

- R(A, B, C, D, E, F)
- S: AB→C, BC→AD, D→E, CF →B

- {A,B}$^+$??
- Is {A,B} a key ??
- How do we find out the key(s) from R??

Example: BCNF Decomposition

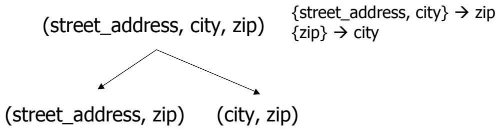
id	name	address	assignment	due	grade
1	John	123 Main St.	HW1	2009-06-22	A-
1	John	123 Main St.	HW2	2009-07-10	B
2	Jane	456 State St.	HW1	2009-06-22	A

class_records

⇩

??

Motivation for 3NF



- We lose the FD {street_address,city}→zip after decomposition, or in other words, it becomes *unenforceable*.

An Unenforceable FD

Before decomposition:

street	city	zip
545 Tech Sq.	Cambridge	02138
545 Tech Sq.	Cambridge	02139

Data error like this can be detected

After decomposition:

street	zip
545 Tech Sq.	02138
545 Tech Sq.	02139

city	zip
Cambridge	02138
Cambridge	02139

The same data error can no longer be detected.

Third Normal Form (3NF)

- A table R is in 3NF if for every nontrivial *FD* **A** → B in R,
  - **A** is a super key of R
  - or B is part of a key of R