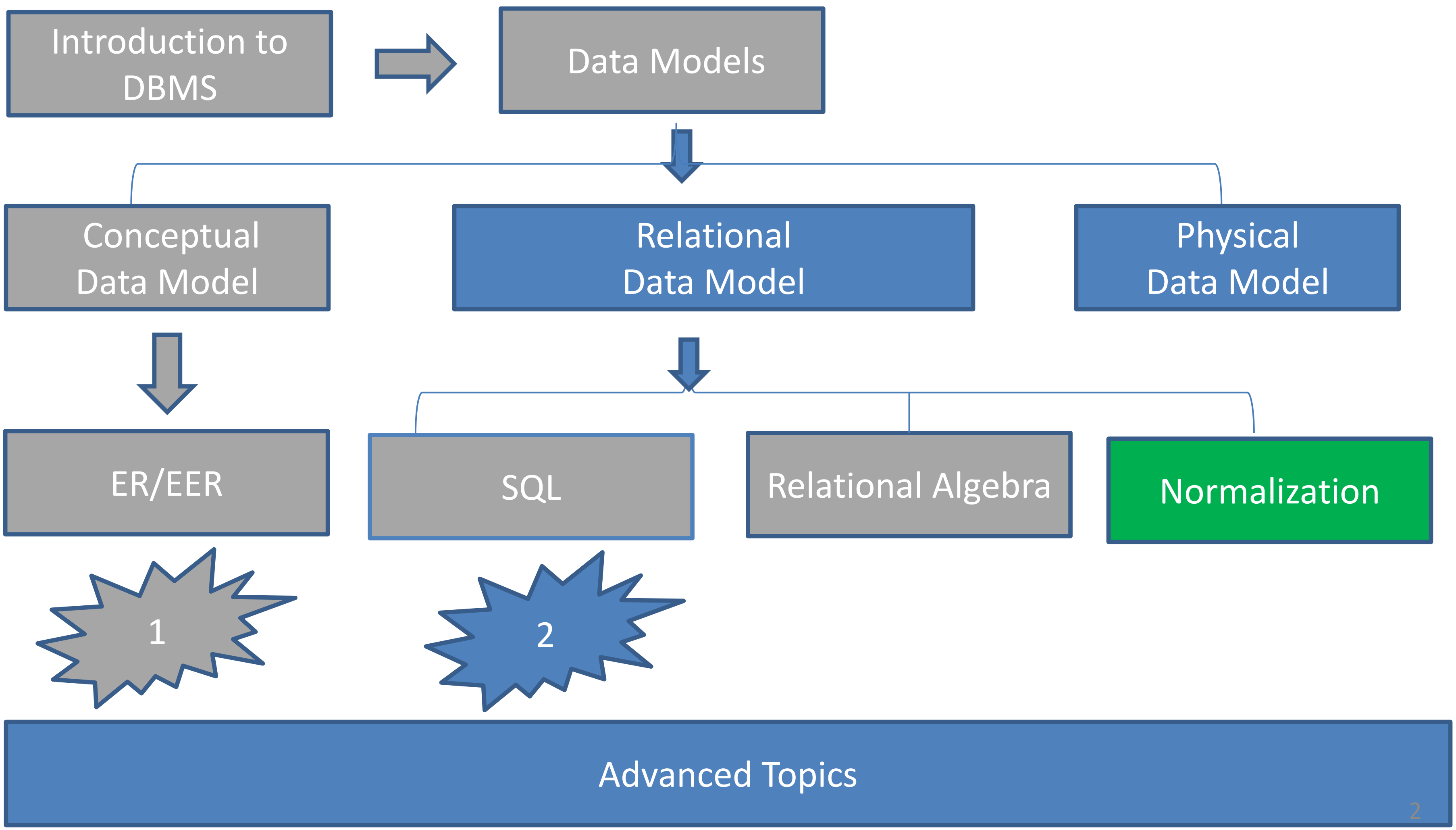
An Introduction to the Database Management Systems

By
Hossein Rahmani

Slides originally by Book(s) Resources

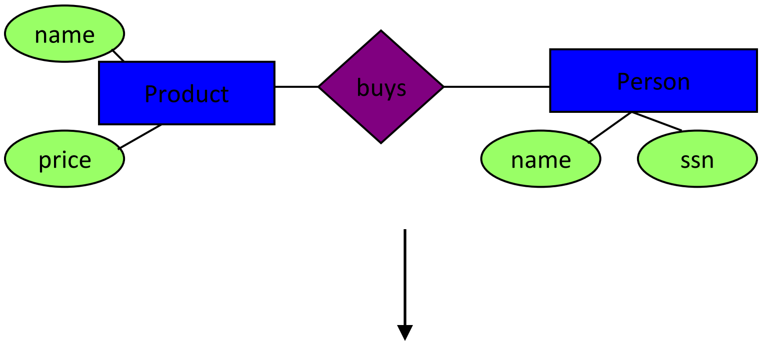
Road Map

(Might change!)

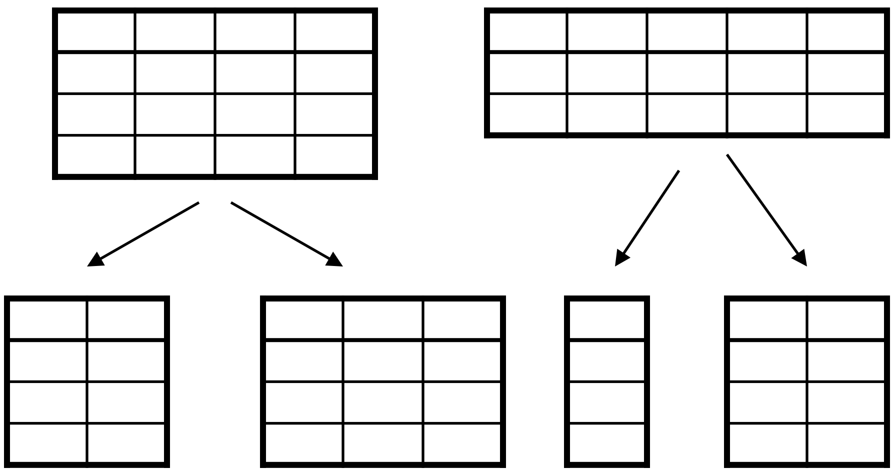


Relational Schema Design

Conceptual Model:



Relational Model:
plus FD's



Normalization:
Eliminates anomalies

Normalization

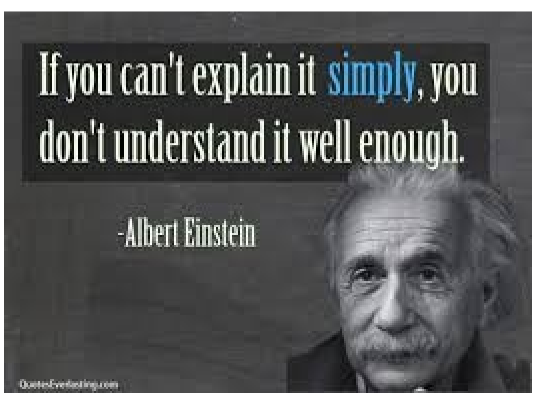
- Informal Design Guidelines for Relation Schemas
- Functional Dependencies
- Normal Forms

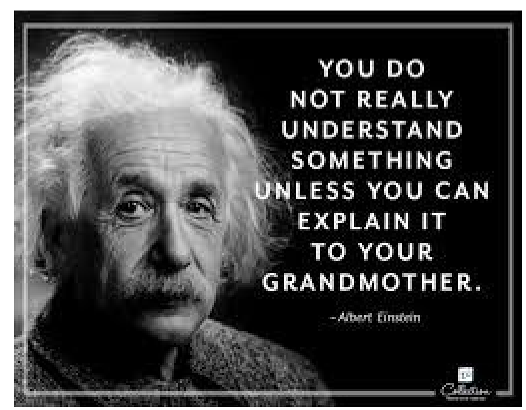
Informal Design Guidelines for Relation Schemas

- Measures of quality
  - Making sure attribute semantics are clear
  - Reducing redundant information in tuples
  - Reducing NULL values in tuples
  - Disallowing possibility of generating spurious tuples

Imparting Clear Semantics to Attributes in Relations

- Semantics of a relation
  - Meaning resulting from interpretation of attribute values in a tuple
- Easier to explain semantics of relation
  - Indicates better schema design

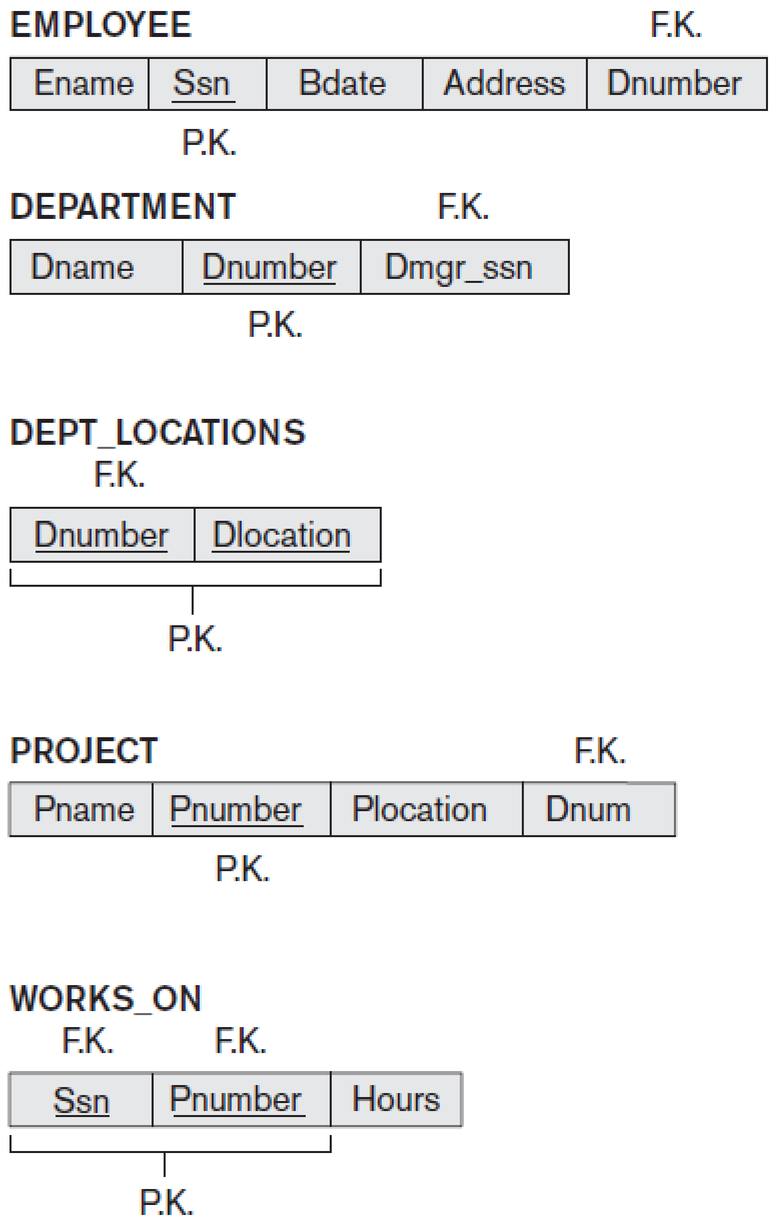




Guideline 1: Semantics

- Make sure that the semantics (meaning) of all base relations and attributes is clear
  - Tuples must be easily interpreted as 'facts'
  - Do not mix, if possible, attributes of more than one entity or relation type in one base relation

A simplified COMPANY relational database schema.



Guideline 1 (cont'd.) Examples of poor design

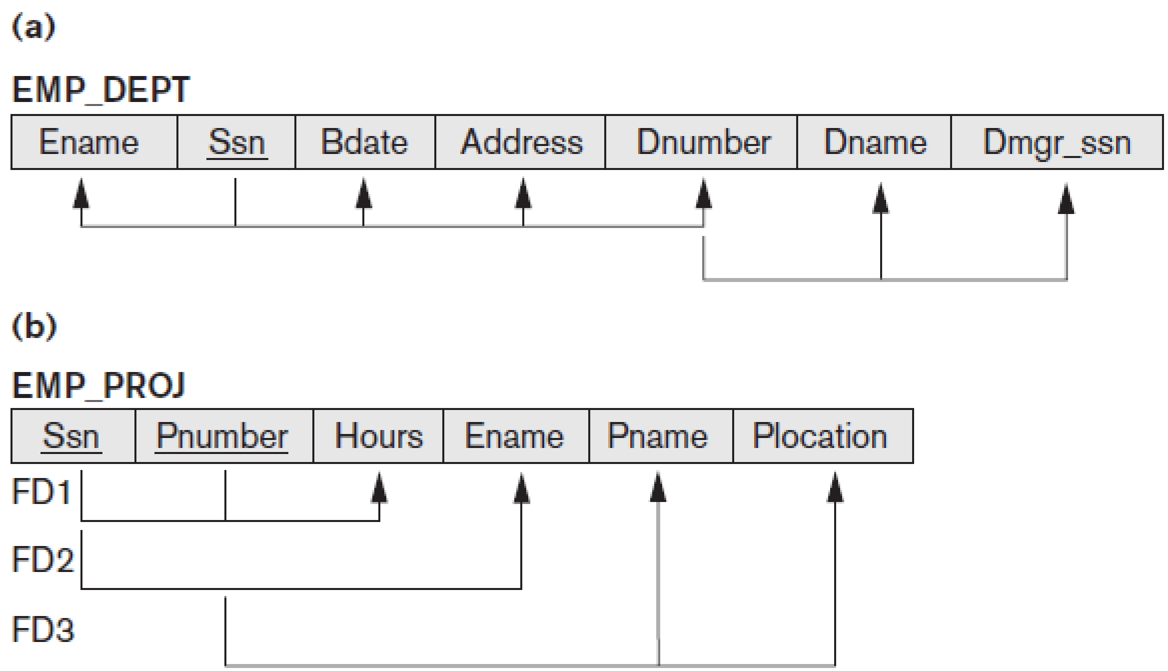


Figure 15.3
Two relation schemas suffering from update anomalies. (a) EMP_DEPT and (b) EMP_PROJ.

Redundant Information in Tuples and Update Anomalies

- Grouping attributes into relation schemas
  - Significant effect on storage space
- Storing natural joins of base relations leads to **update anomalies**
- Types of update anomalies:
  - Insertion
  - Deletion
  - Modification

Redundancy - example

Redundancy

EMP_DEPT

Ename	Ssn	Bdate	Address	Dnumber	Dname	Dmgr_ssn
Smith, John B.	123456789	1965-01-09	731 Fondren, Houston, TX	5	Research	333445555
Wong, Franklin T.	333445555	1955-12-08	638 Voss, Houston, TX	5	Research	333445555
Zelaya, Alicia J.	999887777	1968-07-19	3321 Castle, Spring, TX	4	Administration	987654321
Wallace, Jennifer S.	987654321	1941-06-20	291 Berry, Bellaire, TX	4	Administration	987654321
Narayan, Ramesh K.	666884444	1962-09-15	975 FireOak, Humble, TX	5	Research	333445555
English, Joyce A.	453453453	1972-07-31	5631 Rice, Houston, TX	5	Research	333445555
Jabbar, Ahmad V.	987987987	1969-03-29	980 Dallas, Houston, TX	4	Administration	987654321
Borg, James E.	888665555	1937-11-10	450 Stone, Houston, TX	1	Headquarters	888665555

Guideline 2: Redundancy and anomalies

- **Avoid redundancy**: reduce the space that is needed to store the database as much as possible
- **Prevent anomalies** when changing data in the database
  - update (insertion / deletion / modification) anomalies

Redundancy - example

Redundancy

EMP_DEPT

Ename	Ssn	Bdate	Address	Dnumber	Dname	Dmgr_ssn
Smith, John B.	123456789	1965-01-09	731 Fondren, Houston, TX	5	Research	333445555
Wong, Franklin T.	333445555	1955-12-08	638 Voss, Houston, TX	5	Research	333445555
Zelaya, Alicia J.	999887777	1968-07-19	3321 Castle, Spring, TX	4	Administration	987654321
Wallace, Jennifer S.	987654321	1941-06-20	291 Berry, Bellaire, TX	4	Administration	987654321
Narayan, Ramesh K.	666884444	1962-09-15	975 FireOak, Humble, TX	5	Research	333445555
English, Joyce A.	453453453	1972-07-31	5631 Rice, Houston, TX	5	Research	333445555
Jabbar, Ahmad V.	987987987	1969-03-29	980 Dallas, Houston, TX	4	Administration	987654321
Borg, James E.	888665555	1937-11-10	450 Stone, Houston, TX	1	Headquarters	888665555

Update Anomalies

- Cause: doubly stored data, wrong design

  – insertion anomalies:
    - new tuple contains incorrect attribute value for an already stored entity
    - new entity has a null key
  – deletion anomalies
    - Incomplete deletion of an entity
    - unwanted deletion of an entity
  – modification anomalies
    - incomplete modification of an entity

14

Guideline 3: NULL-values

- Some base relations contain many attributes that often are ‘NULL’
  - Unnecessary use of space
  - Multiple meanings of ‘NULL’
  - JOIN operations can have undesired effects
  - COUNT and SUM can go wrong
- SO: place an attribute in a base relation in which it is as least as possible ‘NULL’

Guideline 4: False (Spurious) Tuples

- If we select base relations wrong, a (NATURAL-)JOIN can create tuples that do not have any connection with the mini world (see next slides)
- So: select base relations such that at a <u>JOIN on primary or foreign keys</u>, no spurious tuples can occur. Don’t JOIN on other attributes

Generation of Spurious Tuples

- Figure (a)
  - Relation schemas EMP_LOCS and EMP_PROJ1
- NATURAL JOIN
  - Result produces many more tuples than the original set of tuples in EMP_PROJ
  - Called **spurious tuples**
  - Represent spurious information that is not valid

Wrong choice - relations

(a)

EMP_LOCS

Ename	Plocation

P.K.

EMP_PROJ1

Ssn	Pnumber	Hours	Pname	Plocation

P.K.

Wrong choice: states

(b)

EMP_LOCS

Ename	Plocation
Smith, John B.	Bellaire
Smith, John B.	Sugarland
Narayan, Ramesh K.	Houston
English, Joyce A.	Bellaire
English, Joyce A.	Sugarland
Wong, Franklin T.	Sugarland
Wong, Franklin T.	Houston
Wong, Franklin T.	Stafford
Zelaya, Alicia J.	Stafford
Jabbar, Ahmad V.	Stafford
Wallace, Jennifer S.	Stafford
Wallace, Jennifer S.	Houston
Borg, James E.	Houston

EMP_PROJ1

Ssn	Pnumber	Hours	Pname	Plocation
123456789	1	32.5	ProductX	Bellaire
123456789	2	7.5	ProductY	Sugarland
666884444	3	40.0	ProductZ	Houston
453453453	1	20.0	ProductX	Bellaire
453453453	2	20.0	ProductY	Sugarland
333445555	2	10.0	ProductY	Sugarland
333445555	3	10.0	ProductZ	Houston
333445555	10	10.0	Computerization	Stafford
333445555	20	10.0	Reorganization	Houston
999887777	30	30.0	Newbenefits	Stafford
999887777	10	10.0	Computerization	Stafford
987987987	10	35.0	Computerization	Stafford
987987987	30	5.0	Newbenefits	Stafford
987654321	30	20.0	Newbenefits	Stafford
987654321	20	15.0	Reorganization	Houston
888665555	20	NULL	Reorganization	Houston

Natural join -> spurious tuples (marked *)

	Ssn	Pnumber	Hours	Pname	Plocation	Ename
	123456789	1	32.5	ProductX	Bellaire	Smith, John B.
*	123456789	1	32.5	ProductX	Bellaire	English, Joyce A.
	123456789	2	7.5	ProductY	Sugarland	Smith, John B.
*	123456789	2	7.5	ProductY	Sugarland	English, Joyce A.
*	123456789	2	7.5	ProductY	Sugarland	Wong, Franklin T.
	666884444	3	40.0	ProductZ	Houston	Narayan, Ramesh K.
*	666884444	3	40.0	ProductZ	Houston	Wong, Franklin T.
*	453453453	1	20.0	ProductX	Bellaire	Smith, John B.
	453453453	1	20.0	ProductX	Bellaire	English, Joyce A.
*	453453453	2	20.0	ProductY	Sugarland	Smith, John B.
	453453453	2	20.0	ProductY	Sugarland	English, Joyce A.
*	453453453	2	20.0	ProductY	Sugarland	Wong, Franklin T.
*	333445555	2	10.0	ProductY	Sugarland	Smith, John B.
*	333445555	2	10.0	ProductY	Sugarland	English, Joyce A.
	333445555	2	10.0	ProductY	Sugarland	Wong, Franklin T.
*	333445555	3	10.0	ProductZ	Houston	Narayan, Ramesh K.
	333445555	3	10.0	ProductZ	Houston	Wong, Franklin T.
	333445555	10	10.0	Computerization	Stafford	Wong, Franklin T.
*	333445555	20	10.0	Reorganization	Houston	Narayan, Ramesh K.
	333445555	20	10.0	Reorganization	Houston	Wong, Franklin T.

⋮

Summary and Discussion of Design Guidelines

- Anomalies cause redundant work to be done
- Waste of storage space due to NULLs
- Difficulty of performing operations and joins due to NULL values
- Generation of invalid and spurious data during joins

Normalization

- Informal Design Guidelines for Relation Schemas
- Functional Dependencies
- Normal Forms

Normalization …

- Using ‘normalization’, you can adhere to these guidelines for a large part
- In a number of steps (algorithms) you transfer a given relational database schema into an ever higher *normal form*
- Base concept: *functional dependency*

Functional Dependencies

- Formal tool for analysis of relational schemas
- Enables us to detect and describe some of the above-mentioned problems in precise terms
- Theory of functional dependency

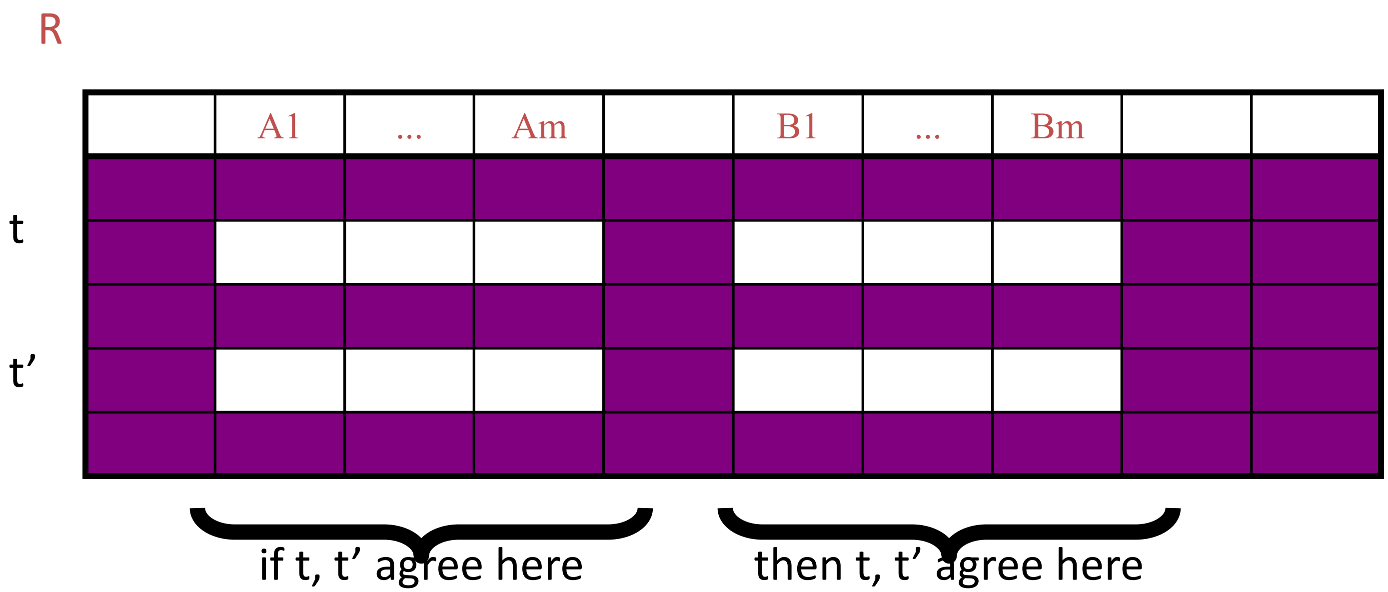
Functional dependency

- Start with one universal relation schema R containing all attributes A1,..,An
- Given two attribute sets X and Y in R
- Functional dependency $X \rightarrow Y$ exists (X functionally determines Y; Y is functionally dependent on X) if:
  - $\forall r(R)$: $\forall t1,t2 \in r$: $t1[X] = t2[X] \rightarrow t1[Y] = t2[Y]$
  - i.e.: component X determines component Y

Functional Dependencies

Definition: $A_1, \ldots, A_m \rightarrow B_1, \ldots, B_n$ holds in R if:

$$\forall t, t' \in R, (t.A_1=t'.A_1 \wedge \ldots \wedge t.A_m=t'.A_m \Rightarrow t.B_1=t'.B_1 \wedge \ldots \wedge t.B_m=t'.B_m)$$



Important Point!

- Functional dependencies are part of the schema!
- They constrain the possible *legal* data instances.
- At any point in time, the actual database may satisfy additional FD's.

Functional dependency

$$AB \rightarrow C$$

<u>A</u>	<u>B</u>	C

Examples

EmpID	Name	Phone	Position
E0045	Smith	1234	Clerk
E1847	John	9876	Salesrep
E1111	Smith	9876	Salesrep
E9999	Mary	1234	Lawyer

- EmpID $\longrightarrow$ Name, Phone, Position
- Position $\longrightarrow$ Phone
- but Phone $\not\longrightarrow$ Position

TABLE_BOOK_DETAIL

Book ID	Genre ID	Genre Type	Price
1	1	Gardening	25.99
2	2	Sports	14.99
3	1	Gardening	10.00
4	3	Travel	12.99
5	2	Sports	17.99

[Book ID] determines-> [Genre ID],
[Genre ID] determines ->[Genre Type]

X	Y	Z
x_1	y_1	z_1
x_1	y_1	z_2
x_2	y_1	z_1
x_2	y_1	z_3

Functional Dependencies?

X	Y	Z
x_1	y_1	z_1
x_1	y_1	z_2
x_2	y_1	z_1
x_2	y_1	z_3

Z → Y,
X → Y,
XZ → Y

- Relation schema S with three attributes ABC
- 3 tuples in a legal instance of a
    - (1,2,3),
    - (4,2,3),
    - (5,3,3).

  - $A \rightarrow B$??
  - $BC \rightarrow A$??
  - $B \rightarrow C$??

- Relation schema S with three attributes ABC
- 3 tuples in a legal instance of a
    - (1,2,3),
    - (4,2,3),
    - (5,3,3).

  - $A \rightarrow B$ <u>YES</u>
  - $BC \rightarrow A$ <u>NO</u>
  - $B \rightarrow C$ <u>YES</u>

Functional dependency

- If X is a superkey of R then $X \rightarrow Y$ holds for each set Y of attributes in R
- If $X \rightarrow Y$, then nothing can be concluded on the existence of $Y \rightarrow X$
- $X \rightarrow Y$ follows from the semantics of the attributes in X and Y (which means that the designer should note and declare it)
- r(R) is *legal* if it agrees with all functional dependencies (FDs) declared on R

Inference rules for FDs

- Six rules for deriving FDs:
  - IR1 (reflexive): if $X \supseteq Y$ then $X \rightarrow Y$ (trivial)

    As a special case: $X \rightarrow X$
  - IR2 (extension): $\{X \rightarrow Y\} \models XZ \rightarrow YZ$
  - IR3 (transitive): $\{X \rightarrow Y, Y \rightarrow Z\} \models X \rightarrow Z$
  - IR4 (project): $\{X \rightarrow YZ\} \models X \rightarrow Y$
  - IR5 (combine): $\{X \rightarrow Y, X \rightarrow Z\} \models X \rightarrow YZ$
  - IR6 (pseudotransitive):

    $\{X \rightarrow Y, WY \rightarrow Z\} \models WX \rightarrow Z$

Inference Rules for FD's

$$A_1, A_2, \ldots A_n \rightarrow B_1, B_2, \ldots B_m$$

Is equivalent to

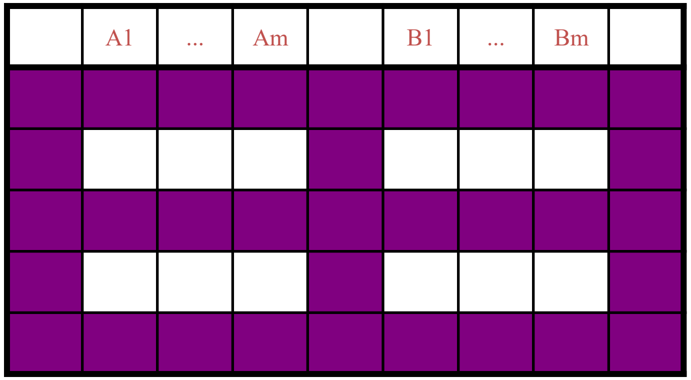
$$A_1, A_2, \ldots A_n \rightarrow B_1$$

$$A_1, A_2, \ldots A_n \rightarrow B_2$$

...

$$A_1, A_2, \ldots A_n \rightarrow B_m$$

Splitting rule
and
Combing rule



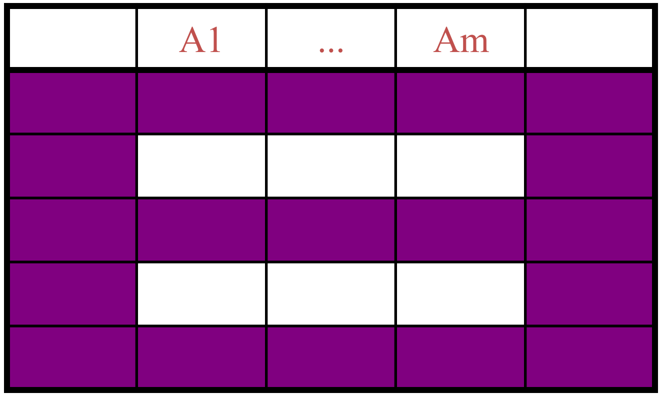
Inference Rules for FD's (continued)

$$A_1, A_2, \ldots A_n \longrightarrow A_i$$

Trivial Rule

where i = 1, 2, ..., n

Why ?



Inference Rules for FD's (continued)

Transitive Closure Rule

If $A_1, A_2, \ldots A_n \longrightarrow B_1, B_2 \ldots, B_m$

and $B_1, B_2, \ldots B_m \longrightarrow C_1, C_2 \ldots, C_p$

then $A_1, A_2, \ldots A_n \longrightarrow C_1, C_2 \ldots, C_p$

Why ?

	A1	...	Am		B1	...	Bm		C_1	...	C_p	

Closure F^+ of F

- Given a set of FDs for R: F(R)
- IR1-3 is *sound* & *complete* (Armstrong)
  - Sound: If a new FD f can be derived from F(R) using IR1-3, and r(R) is legal for F, then r(R) is also legal for F $\cup$ {f}
  - Complete: If FD f holds on R then f can be derived from F using IR1-3
- The <u>set F^+(R) of all FDs that can be derived from F</u>, is called the closure F^+ of F(R)

- R=(A, B, C, G, H, I)
- FD = { A→B, A→C, CG→H, CG→I, B→H }
- Find F**+**
  - A→H (transitivity rule)
  - CG→HI (union rule)
  - AG→I, Since A→C & CG→I (pseudotransitivity)

Closure X^+ under F

- Given $X \rightarrow Y \in F$. The *closure* X^+ of X under F is the set of all attributes that are also functionally dependent on X
- **Closure** of a set of attributes X with respect to F is the set X+ of <u>all attributes that are functionally determined by X</u>

Algorithm to determine X^+:

$$
\begin{array}{l}
X^+ := X\ ; \\
\text{do } \{\ \ \text{old}X^+ = X^+\ ; \\
\qquad \text{for all } Y \rightarrow Z \in F : \\
\qquad\qquad \text{if } X+ \supseteq Y \text{ then } X^+ = X^+ \cup Z; \\
\}\ \text{while } (\text{old}X^+ \neq X^+)
\end{array}
$$

Closure of Attribute Sets

- R=(A, B, C, G, H, I)
- FD = { A→B, A→C, CG→H, CG→I, B→H }
- Find (AG)**+**
  - A→B causes us to include B in result.
  - A→C causes result to become ABCG.
  - CG→H causes result to become ABCGH.
  - CG→I causes result to become ABCGHI.

Equivalence

- Two sets of FDs, F and E are equivalent ($F \cong E$) iff $F^+ = E^+$
- Semantically: if $F \cong E$, then r(R) is legal for F *iff* r(R) is legal for E
- By definition: $F \models f$ *iff* $F \cong F \cup \{f\}$
- For each set F there exist many equivalent sets of FDs. We prefer *simplicity*: minimal cover

Minimal Cover

- We can translate any F into an equivalent minimal cover G
- A set FDs G is a *minimal cover* of F if $G \cong F$ and
  - for all $X \rightarrow Y \in G$, Y has exactly one attribute (so, if $X \rightarrow YZ$, then split into $X \rightarrow Y$ and $X \rightarrow Z$)
  - We cannot remove any $X \rightarrow Y$ from G without loosing equivalence with F
  - We cannot replace any $X \rightarrow Y$ in G by $W \rightarrow Y$ with $W \subset X$, without loosing equivalence with F

46

Algorithm for Minimal Cover

1) Start with $G := F$;
2) Replace all $X \rightarrow Y$ with $Y = \{A_1,..,A_n\}$ by $X \rightarrow A_i$; (IR4)
3) For all $XY \rightarrow A$: if $G - \{XY \rightarrow A\} \cong G \cup \{X \rightarrow A\}$
   then replace $XY \rightarrow A$ with $X \rightarrow A$;
4) If $G - \{X \rightarrow A\} \cong G$ then remove $X \rightarrow A$;

Example:

1) $AB \rightarrow CD$; $C \rightarrow D$; $A \rightarrow CB$;
2) $AB \rightarrow C$; $AB \rightarrow D$; $C \rightarrow D$; $A \rightarrow C$; $A \rightarrow B$;
3) $A \rightarrow C$; $A \rightarrow D$; $C \rightarrow D$; $A \rightarrow B$;
4) $A \rightarrow C$; $C \rightarrow D$; $A \rightarrow B$;

Quiz 1

- Determine the functional dependencies?

TEACH

Teacher	Course	Text
Smith	Data Structures	Bartram
Smith	Data Management	Martin
Hall	Compilers	Hoffman
Brown	Data Structures	Horowitz

Quiz 2

- Consider a relation *R* with five attributes *ABCDE*.
- For each of the following instances of R, state whether it violates the FD $BC \rightarrow D$???
  - (a) *{ }* (i.e., empty relation)
  - (b) *{*(*a*,2,3,4,5), (2,*a*,3,5,5)*}*
  - (c) *{*(*a*,2,3,4,5), (2,*a*,3,5,5), (*a*,2,3,4,6)*}*
  - (d) *{*(*a*,2,3,4,5), (2,*a*,3,4,5), (*a*,2,3,6,5)*}*
  - (e) *{*(*a*,2,3,4,5), (2,*a*,3,7,5), (*a*,2,3,4,6)*}*
  - (f) *{*(*a*,2,3,4,5), (2,*a*,3,4,5), (*a*,2,3,6,5), (*a*,2,3,6,6)*}*
  - (g) *{*(*a*,2,3,4,5), (*a*,2,3,6,5), (*a*,2,3,6,6), (*a*,2,3,4,6)*}*

Quiz 3

- $E : \{B \rightarrow A, D \rightarrow A, AB \rightarrow D\}$.

- Find the minimal cover?