An Introduction to the Database Management Systems

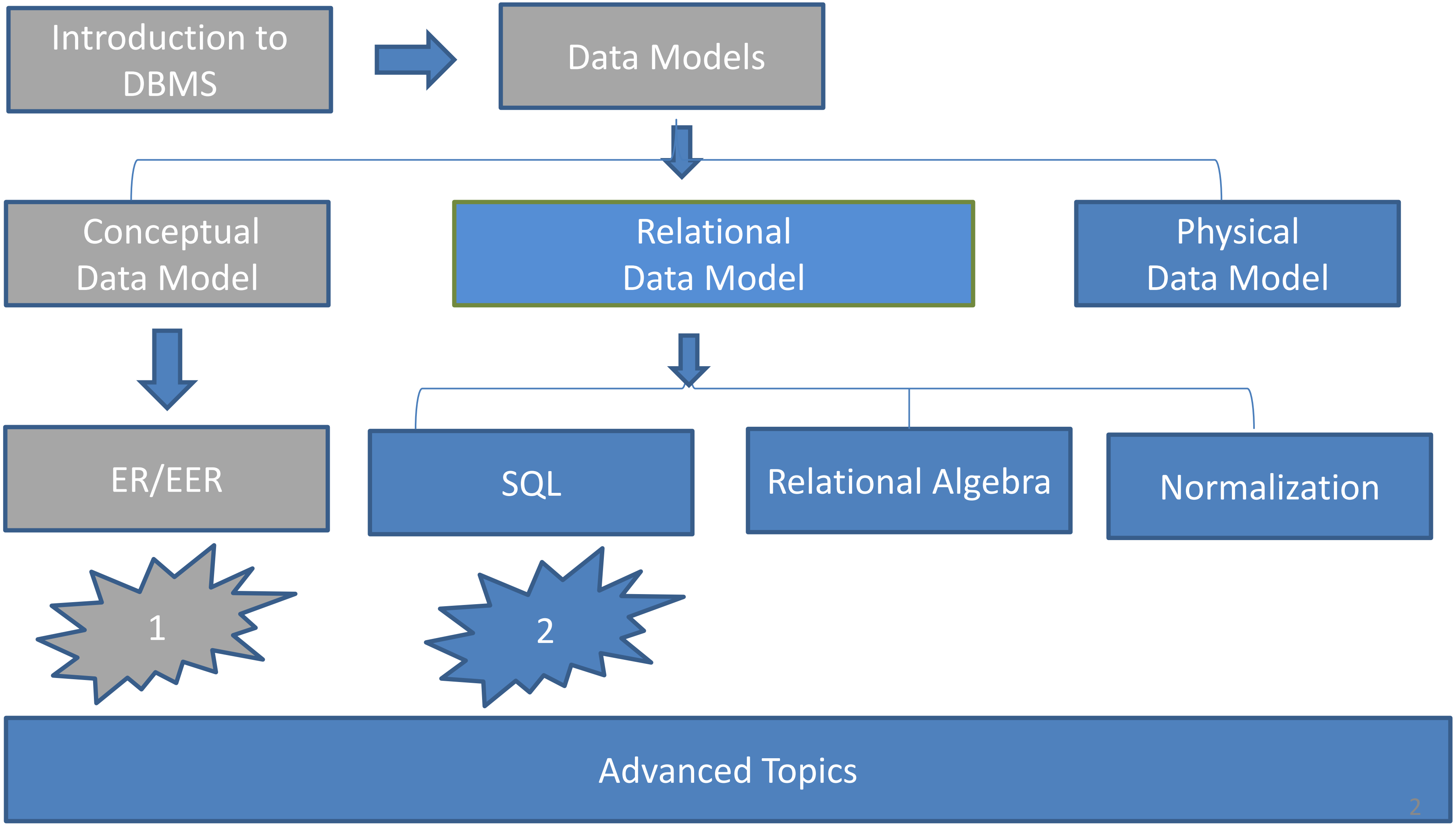
By

Hossein Rahmani

Slides originally by Book(s) Resources

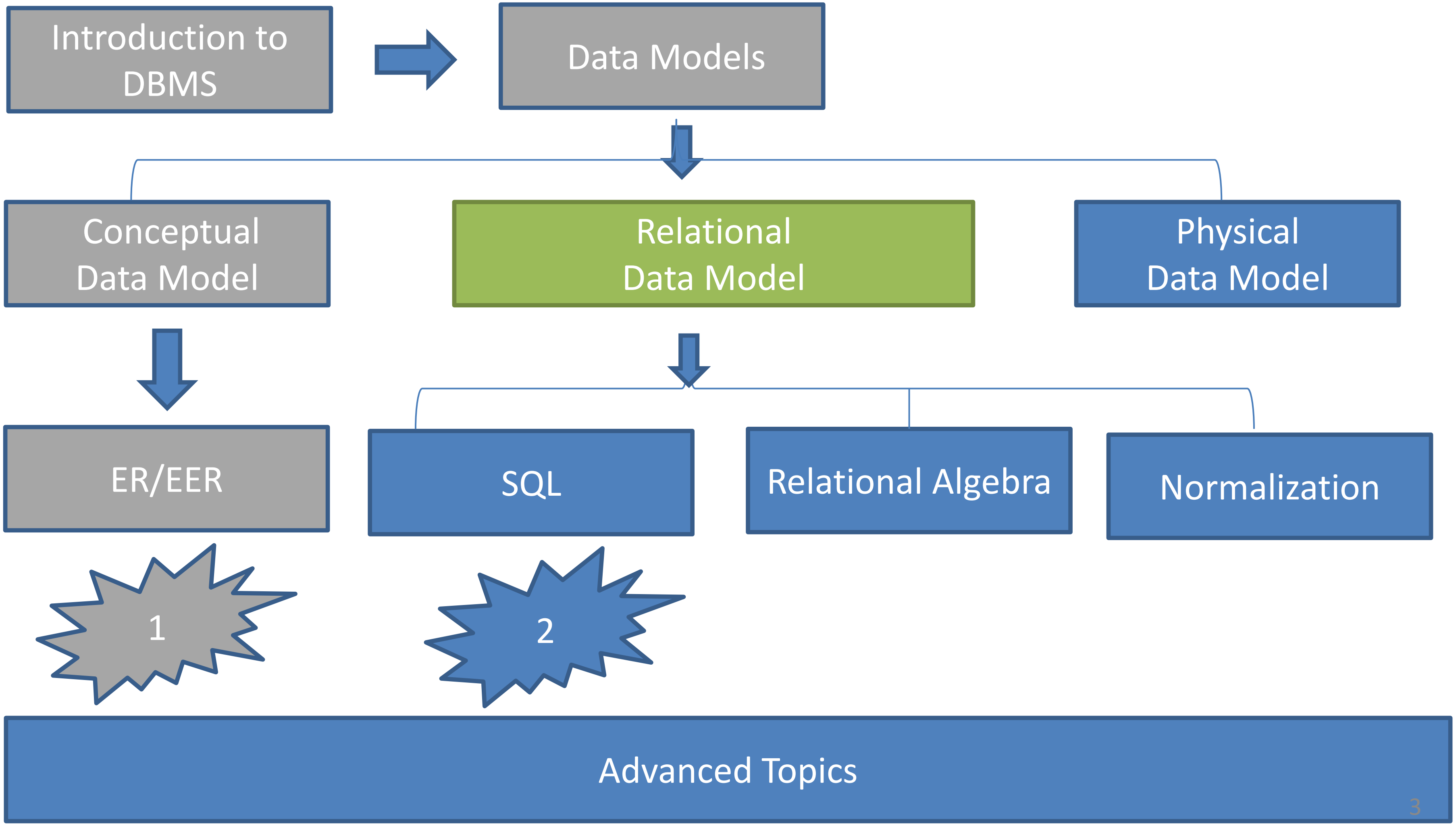
Road Map

(Might change!)



Road Map

(Might change!)



Introduction to DBMS

Data Models

Conceptual Data Model

Relational Data Model

Physical Data Model

ER/EER

SQL

Relational Algebra

Normalization

1

2

Advanced Topics

(Previously Discussed) Data models

- Conceptual (high-level)
  - for end users / analists

versus

- Physical (low-level)
  - for programmers

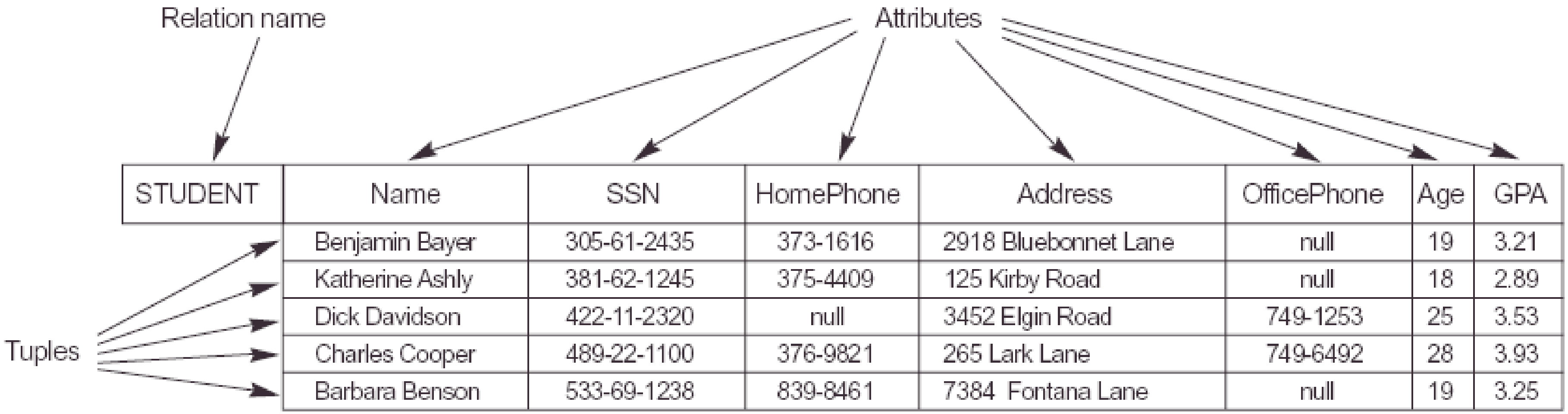
- In between: Representational (implementation) data model

Relational Model

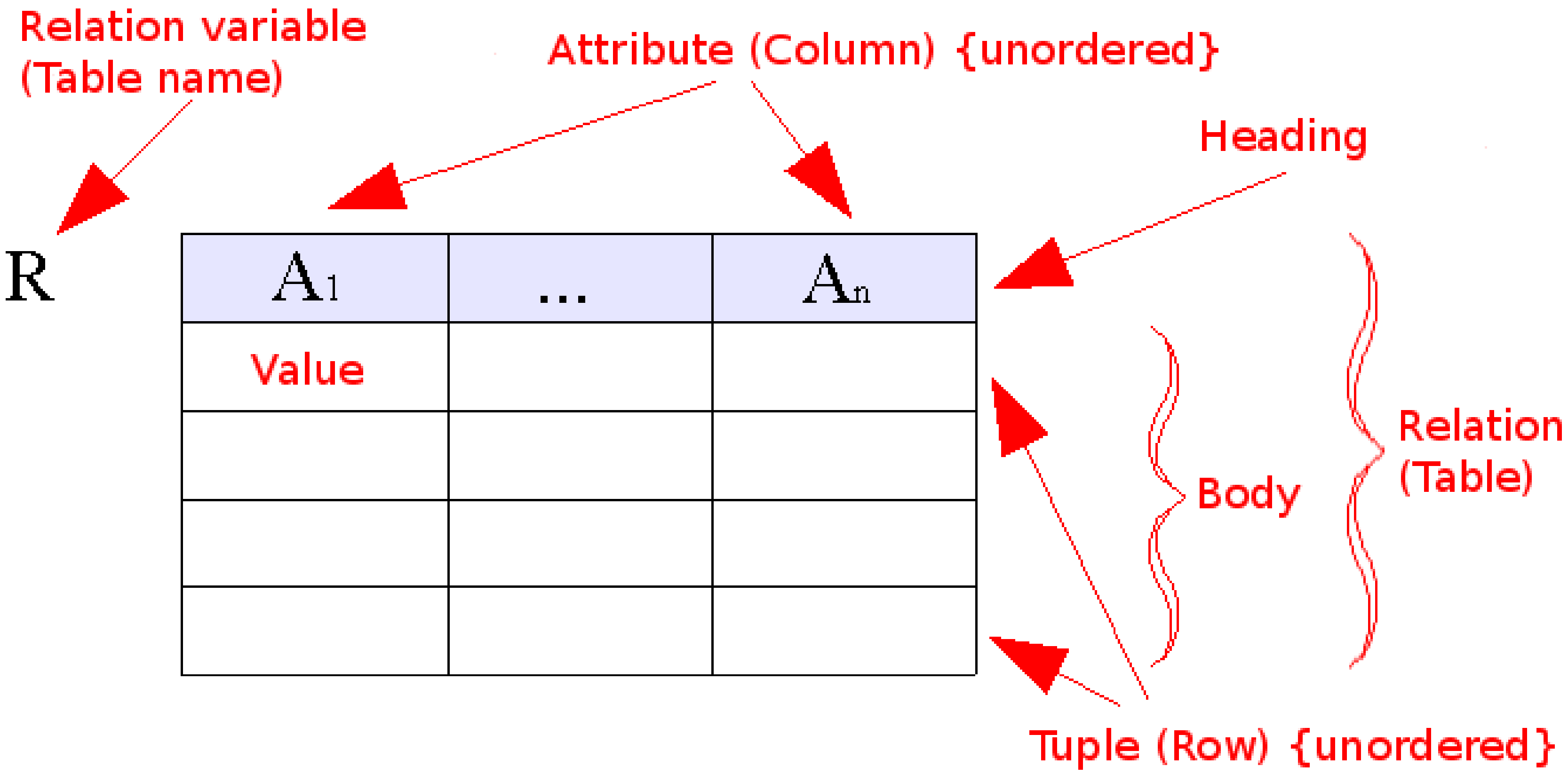
- We now move from the *conceptual* level to the *representational* (in-between) level: how do we represent databases (in a DBMS)?

- A database is represented by *relations*
- A relation is a table of which the columns are called *attributes* and the rows *tuples*
- Attributes each have a *domain*

Relations, attributes, tuples



STUDENT	Name	SSN	HomePhone	Address	OfficePhone	Age	GPA
	Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	null	19	3.21
	Katherine Ashly	381-62-1245	375-4409	125 Kirby Road	null	18	2.89
	Dick Davidson	422-11-2320	null	3452 Elgin Road	749-1253	25	3.53
	Charles Cooper	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
	Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	null	19	3.25



Relation variable (Table name)

Attribute (Column) {unordered}

Heading

R

A_1	...	A_n
Value		

Body

Relation (Table)

Tuple (Row) {unordered}

Domain

- A domain is a set of atomic values (so not composite and/or multi-valued):
  - Phone numbers in the Netherlands
  - Ages
  - Student-ids
- A domain is determined by:
  - Name
  - Data type
  - Format

Schema and diagram

- A *schema* in database theory is a formal (logical or mathematical) description of the *form* (e.g. a database or a relation)
- A *diagram* is the graphical representation of a schema

Relation schema

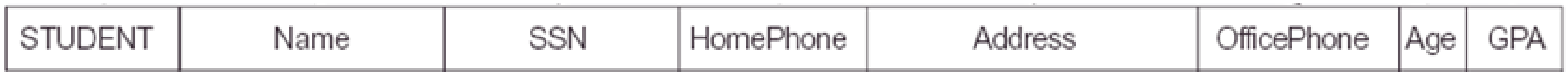
- A relation schema gives the form of a relation (intention)
- A relation schema $R(A_1,A_2,A_3,\ldots,A_n)$ of a relation of degree n consists of the name R of the relation and a list of n attributes $A_1,A_2,A_3,\ldots,A_n$
- An attribute A_i is the name of the role of a domain D_i in the relation
- D_i is the domain of A_i and is indicated by $dom(A_i)$

Relation schema

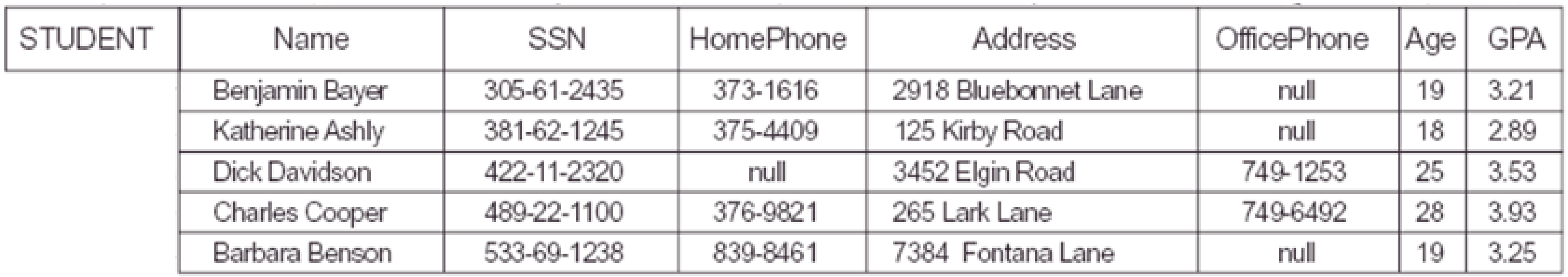
- Example:
  - Student(name, ssn, homephone, address,
    officephone, age, gpa)
  - Domains:
    - Dom(name) = names, e.g. string[40]
    - Dom(ssn) = social-security numbers, e.g. 10-digit integer
    - Dom(homephone) = local-phone-numbers, e.g. …
    - …

Relation (State)

- A relation r or r(R) (relation state or relation instance) of a relation schema $R(A_1,A_2,A_3,\ldots,A_n)$ is a *set of* n-*tuples* $r = \{t_1,t_2,\ldots,t_m\}$.
- Each n-tuple t in r (R) is an ordered *list* of values $t = \langle v_1,v_2,\ldots,v_n \rangle$ in which each value v_i either is member of $dom(A_i)$ or the special value *null*



STUDENT	Name	SSN	HomePhone	Address	OfficePhone	Age	GPA



STUDENT	Name	SSN	HomePhone	Address	OfficePhone	Age	GPA
	Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	null	19	3.21
	Katherine Ashly	381-62-1245	375-4409	125 Kirby Road	null	18	2.89
	Dick Davidson	422-11-2320	null	3452 Elgin Road	749-1253	25	3.53
	Charles Cooper	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
	Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	null	19	3.25

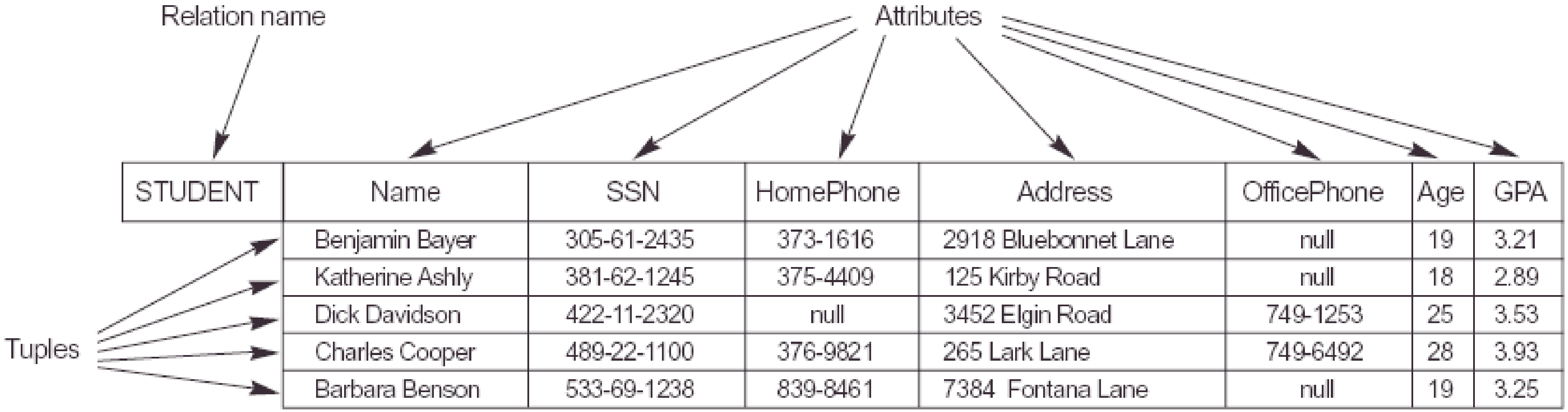
Relations

- A relation r(R) is a mathematical relation of degree *n* on the domains dom(A_1) .. dom(A_n), or:
- r (R) $\subseteq$ dom(A_1) $\times$ dom(A_2) .. $\times$ dom(A_n)
- The number of possible tuples in a relation r(R) is: |dom(A_1)|.|dom(A_2)|. ⋯ .|dom(A_n)|

Properties of relations

- A relation is a **set** of tuples, so:
  - No ordering among the tuples
  - No two tuples can exist with the same values for all attributes
- Tuples are an ordered list: so, the ordering of the attributes is important!
- Attribute values are atomic

Relations, attributes, tuples



STUDENT	Name	SSN	HomePhone	Address	OfficePhone	Age	GPA
	Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	null	19	3.21
	Katherine Ashly	381-62-1245	375-4409	125 Kirby Road	null	18	2.89
	Dick Davidson	422-11-2320	null	3452 Elgin Road	749-1253	25	3.53
	Charles Cooper	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
	Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	null	19	3.25

Interpretations of relations

- Some relations contain facts on entities in the mini world, others on relationships between entities
- A relation schema can also be seen as a predicate with tuples as truth-values that satisfy the predicate

ER Model

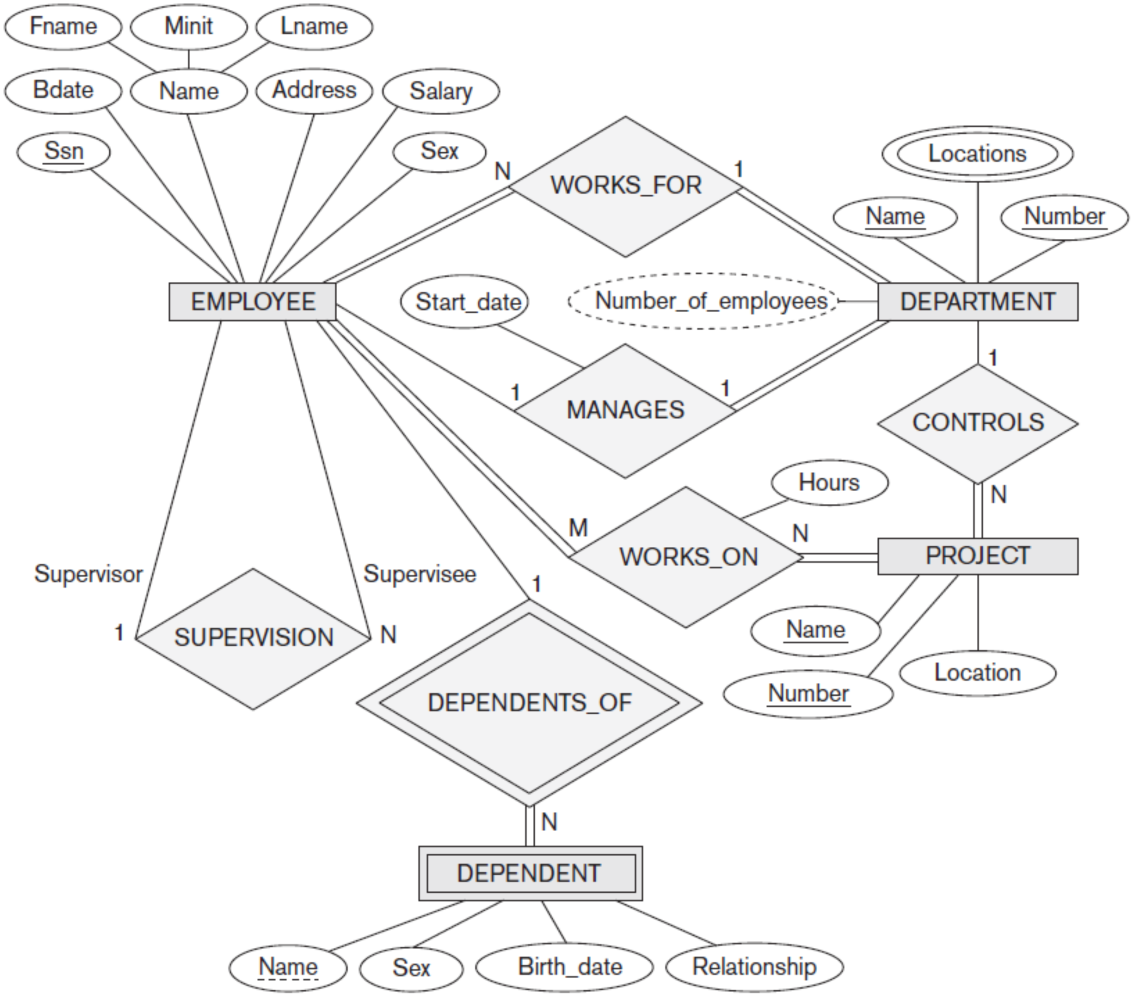
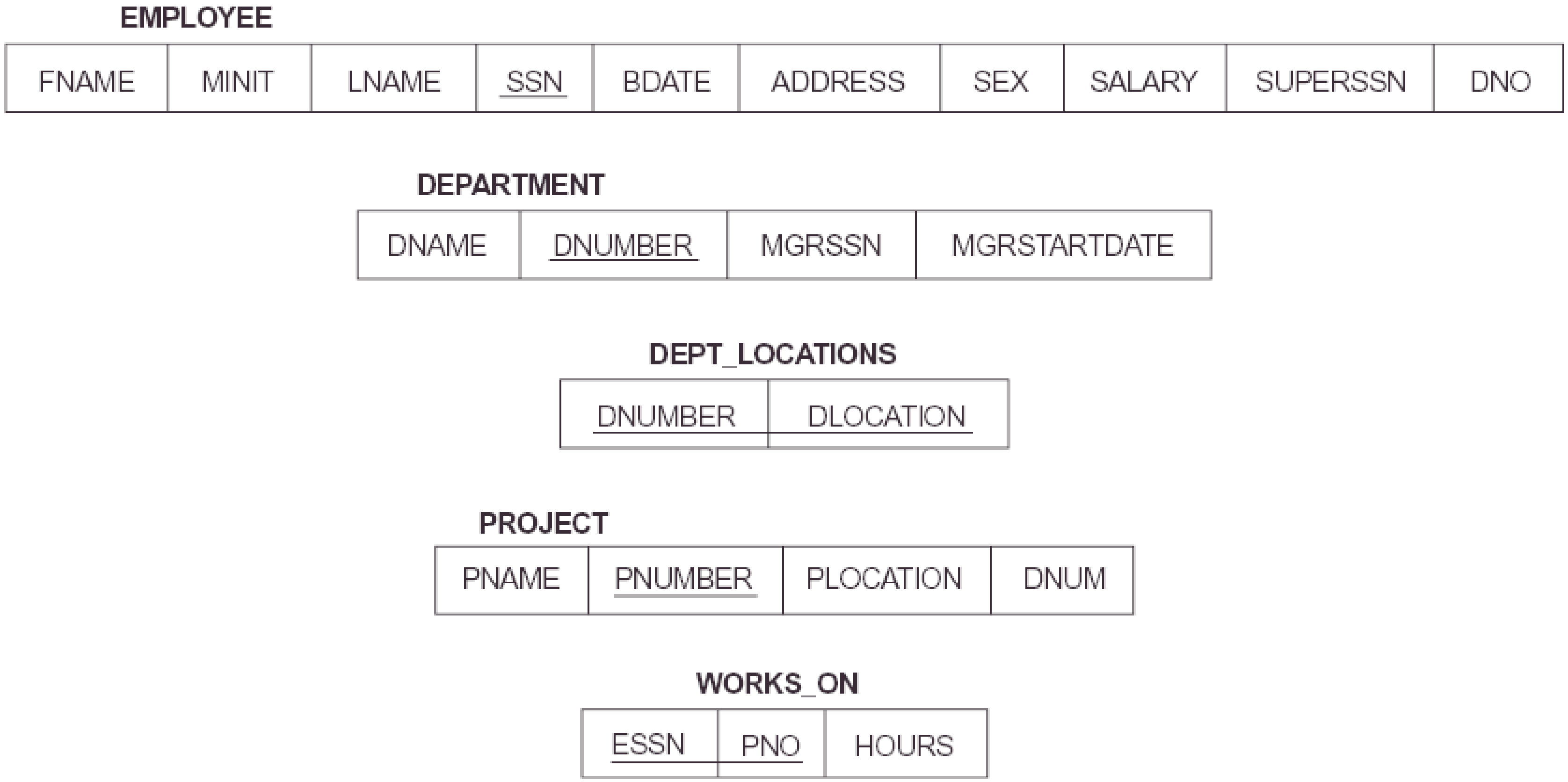


Figure 7.2
An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter and is summarized in Figure 7.14.

Relational Model



EMPLOYEE

FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO

DEPARTMENT

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE

DEPT_LOCATIONS

DNUMBER	DLOCATION

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM

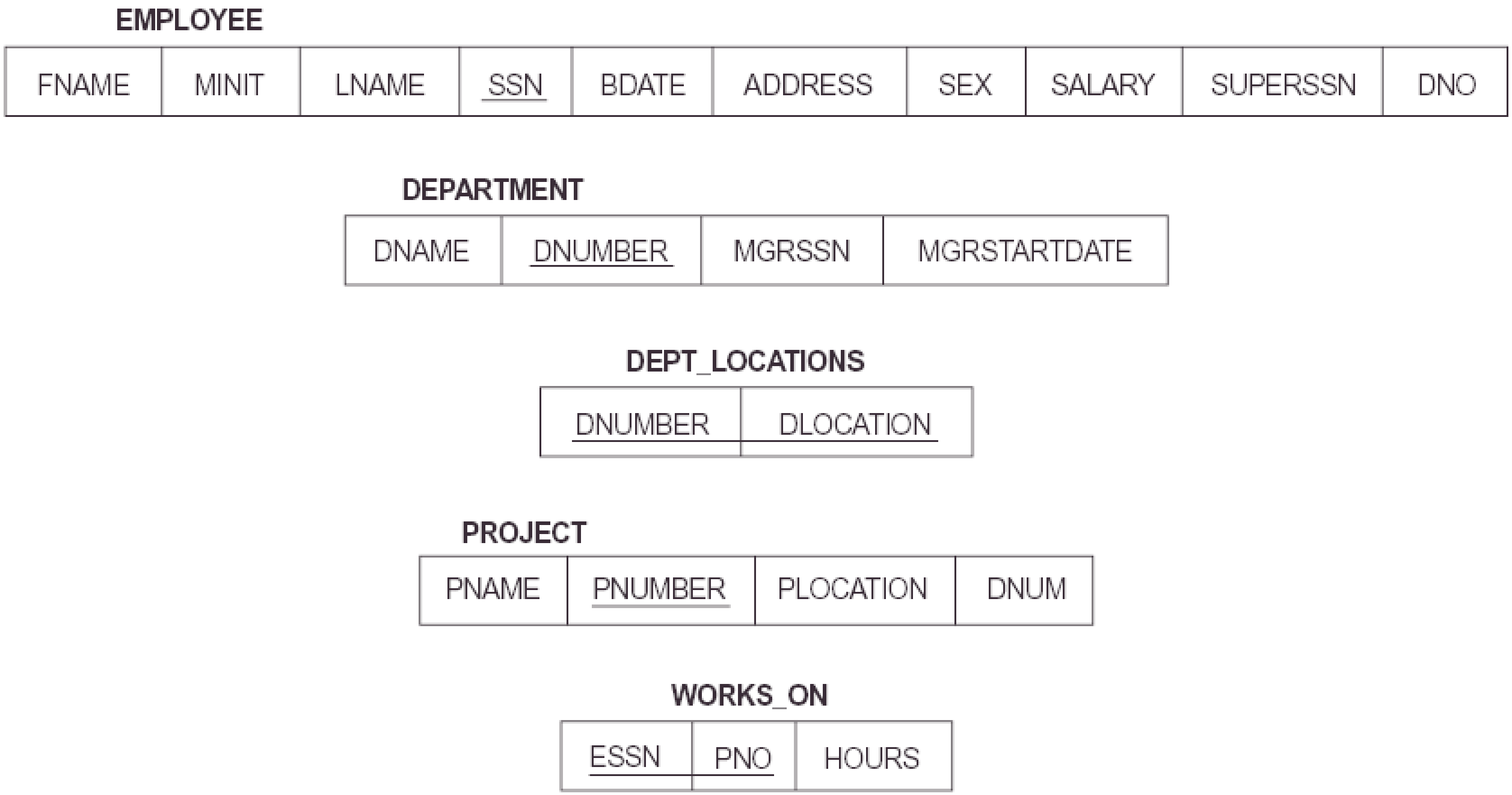
WORKS_ON

ESSN	PNO	HOURS

Relational database schemas

- A relational database **schema** S is a set of relation schemas S = {R_1, R_2, … R_k} and a set of integrity constraints IC
- A relational database **state** DB is a set of relation states {$r_1(R_1)$, $r_2(R_2)$, …, $r_k(R_k)$} such that each r_i fulfills the integrity constraints in IC

Example: Relational Database Schema



EMPLOYEE

FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO

DEPARTMENT

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE

DEPT_LOCATIONS

DNUMBER	DLOCATION

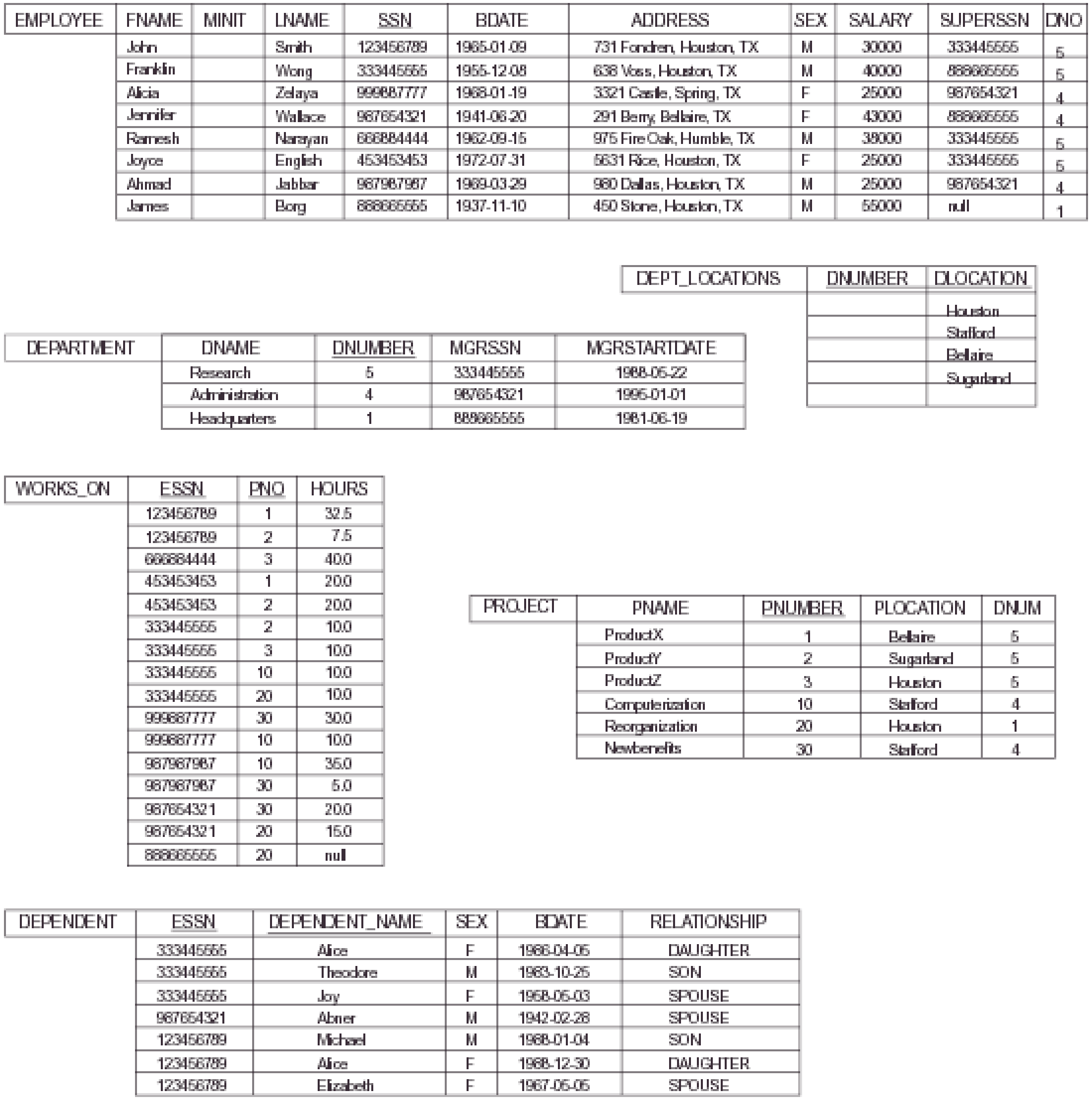
PROJECT

PNAME	PNUMBER	PLOCATION	DNUM

WORKS_ON

ESSN	PNO	HOURS

Example: Relational Database State



EMPLOYEE	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
	John		Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
	Franklin		Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
	Alicia		Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
	Jennifer		Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
	Ramesh		Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
	Joyce		English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
	Ahmad		Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
	James		Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	null	1

DEPT_LOCATIONS	DNUMBER	DLOCATION
		Houston
		Stafford
		Bellaire
		Sugarland

DEPARTMENT	DNAME	DNUMBER	MGRSSN	MGRSTARTDATE
	Research	5	333445555	1988-05-22
	Administration	4	987654321	1995-01-01
	Headquarters	1	888665555	1981-06-19

WORKS_ON	ESSN	PNO	HOURS
	123456789	1	32.5
	123456789	2	7.5
	666884444	3	40.0
	453453453	1	20.0
	453453453	2	20.0
	333445555	2	10.0
	333445555	3	10.0
	333445555	10	10.0
	333445555	20	10.0
	999887777	30	30.0
	999887777	10	10.0
	987987987	10	35.0
	987987987	30	5.0
	987654321	30	20.0
	987654321	20	15.0
	888665555	20	null

PROJECT	PNAME	PNUMBER	PLOCATION	DNUM
	ProductX	1	Bellaire	5
	ProductY	2	Sugarland	5
	ProductZ	3	Houston	5
	Computerization	10	Stafford	4
	Reorganization	20	Houston	1
	Newbenefits	30	Stafford	4

DEPENDENT	ESSN	DEPENDENT_NAME	SEX	BDATE	RELATIONSHIP
	333445555	Alice	F	1986-04-05	DAUGHTER
	333445555	Theodore	M	1983-10-25	SON
	333445555	Joy	F	1958-05-03	SPOUSE
	987654321	Abner	M	1942-02-28	SPOUSE
	123456789	Michael	M	1988-01-04	SON
	123456789	Alice	F	1988-12-30	DAUGHTER
	123456789	Elizabeth	F	1967-05-05	SPOUSE

Constraints

- Three types of constraints
  - 1) inherent (model-based) constraints or implicit constraints
  - 2) constraints in the database model (specified in the DDL): schema-based or explicit constraints
  - 3) other constraints: application-based or semantic constraints (business rules)

Examples of constraints (1st category)

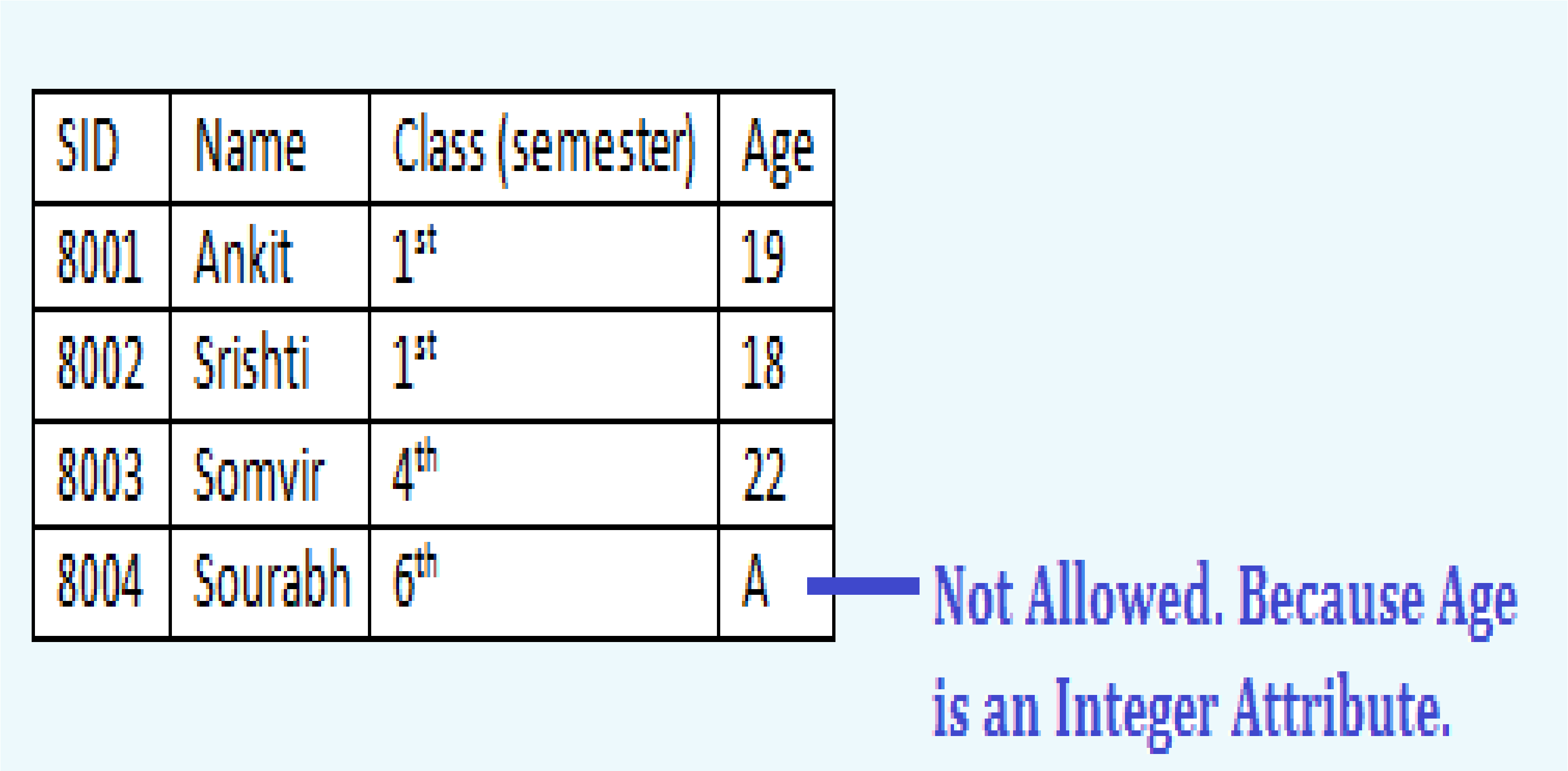
- Attribute names within a relation schema must be different
- All tuples within a relation state must be different
- …
- They are inherent in the data model

Integrity constraints (2nd category)

- Domain constraints
  - Given by dom(A_i)
- Key constraints
- Null constraints
  - (which attribute can/cannot be null)
- Entity-integrity constraints
- Referential integrity constraints

Domain Constraints

- Typically include:
  - Numeric data types for integers and real numbers
  - Characters
  - Booleans
  - Fixed-length strings
  - Variable-length strings
  - Date, time, timestamp
  - Money
  - Other special data types



SID	Name	Class (semester)	Age
8001	Ankit	1st	19
8002	Srishti	1st	18
8003	Somvir	4th	22
8004	Sourabh	6th	A

Not Allowed. Because Age is an Integer Attribute.

Integrity constraints (2^{nd} category)

- Domain constraints
  - Given by dom(A_i)
- Key constraints
- Null constraints
  - (which attribute can/cannot be null)
- Entity-integrity constraints
- Referential integrity constraints

Key constraints

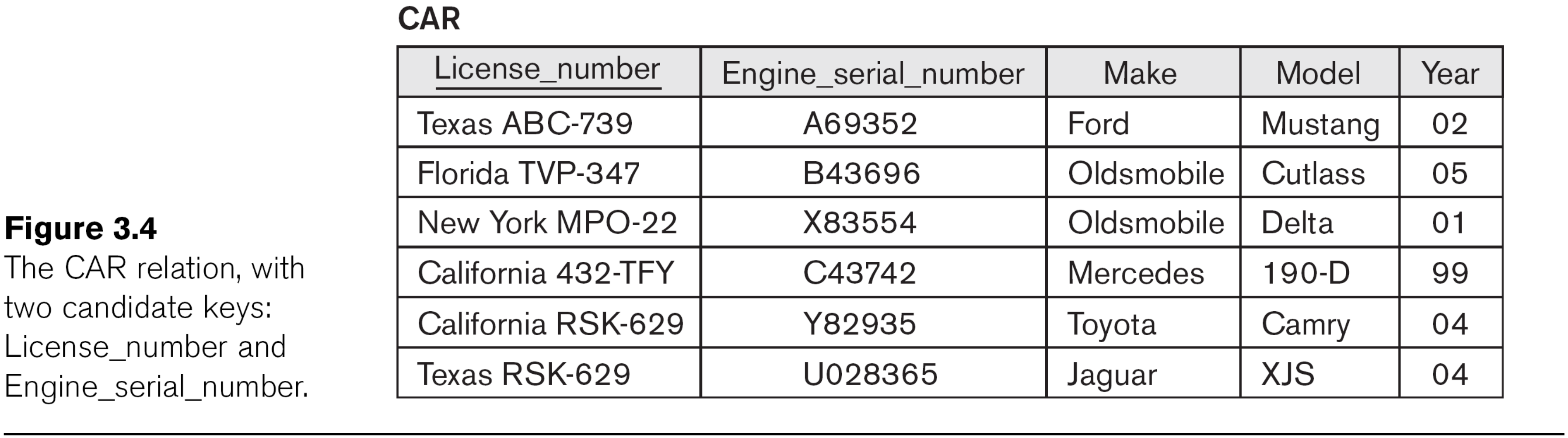
- **Superkey**: Subset SK of attributes of a relation schema for which holds that no two different tuples can have equal values:

  $t_1[SK] = t_2[SK]$ iff $t_1 = t_2$

- A **key** K is a *minimal* superkey: you cannot remove an attribute from K without loosing the superkey property

Key constraints

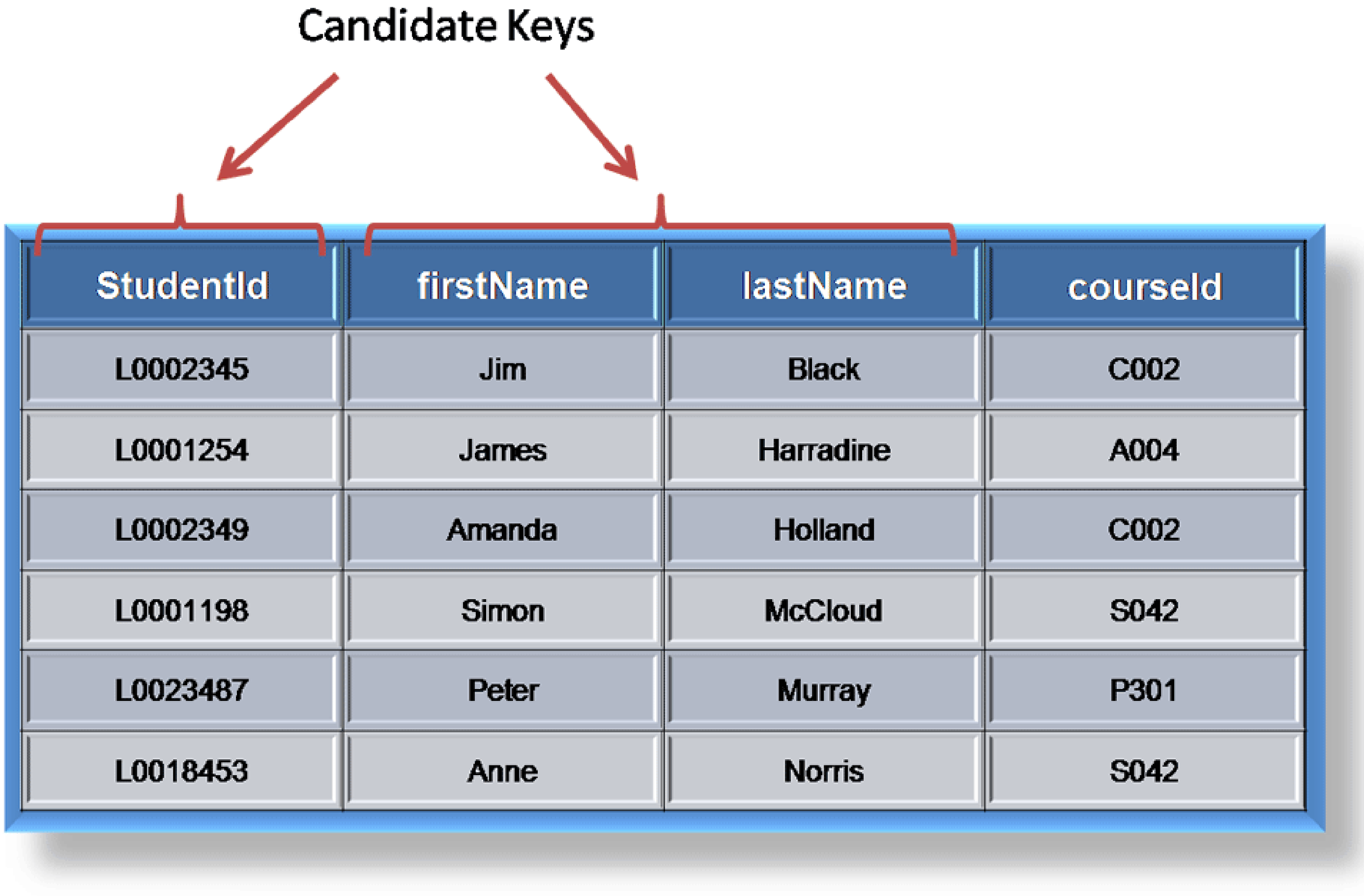
- A relation schema can have more than one key. Each key is a **candidate key**
- One of the candidate keys is denoted as **primary key**
- The attributes in the primary key identify the tuples
- In the relation schema we underline the attributes of the primary key



CAR

License_number	Engine_serial_number	Make	Model	Year
Texas ABC-739	A69352	Ford	Mustang	02
Florida TVP-347	B43696	Oldsmobile	Cutlass	05
New York MPO-22	X83554	Oldsmobile	Delta	01
California 432-TFY	C43742	Mercedes	190-D	99
California RSK-629	Y82935	Toyota	Camry	04
Texas RSK-629	U028365	Jaguar	XJS	04

Figure 3.4
The CAR relation, with two candidate keys: License_number and Engine_serial_number.



Candidate Keys

StudentId	firstName	lastName	courseId
L0002345	Jim	Black	C002
L0001254	James	Harradine	A004
L0002349	Amanda	Holland	C002
L0001198	Simon	McCloud	S042
L0023487	Peter	Murray	P301
L0018453	Anne	Norris	S042

Integrity constraints (2nd category)

- Domain constraints
  - Given by dom(A_i)
- Key constraints
- Null constraints
  - (which attribute can/cannot be null)
- Entity-integrity constraints
- Referential integrity constraints

Integrity constraints (2nd category)

- Domain constraints
  - Given by dom(A_i)
- Key constraints
- Null constraints
  - (which attribute can/cannot be null)
- Entity-integrity constraints
- Referential integrity constraints

Entity-integrity constraint

- The primary key may never be null in a tuple
- The primary key identifies the tuple
- Be aware: a tuple does not always represent an entity!

Entity Constraints

EMPLOYEE

ENO	Name	Age	DNO
1	Ankit	19	10
2	Srishti	18	11
3	Somvir	22	10
4	Sourabh	19	10

DEPARTMENT

DNO	D.Location
10	Rohtak
11	Bhiwani
12	Hansi

NULL	Kavya	21	10

Insertion into EMPLOYEE table is not allowed, Because this insertion violates the Entity Integrity constraints or Integrity Rule 1 as the primary key(ENO) cannot contain a null value.

Integrity constraints (2nd category)

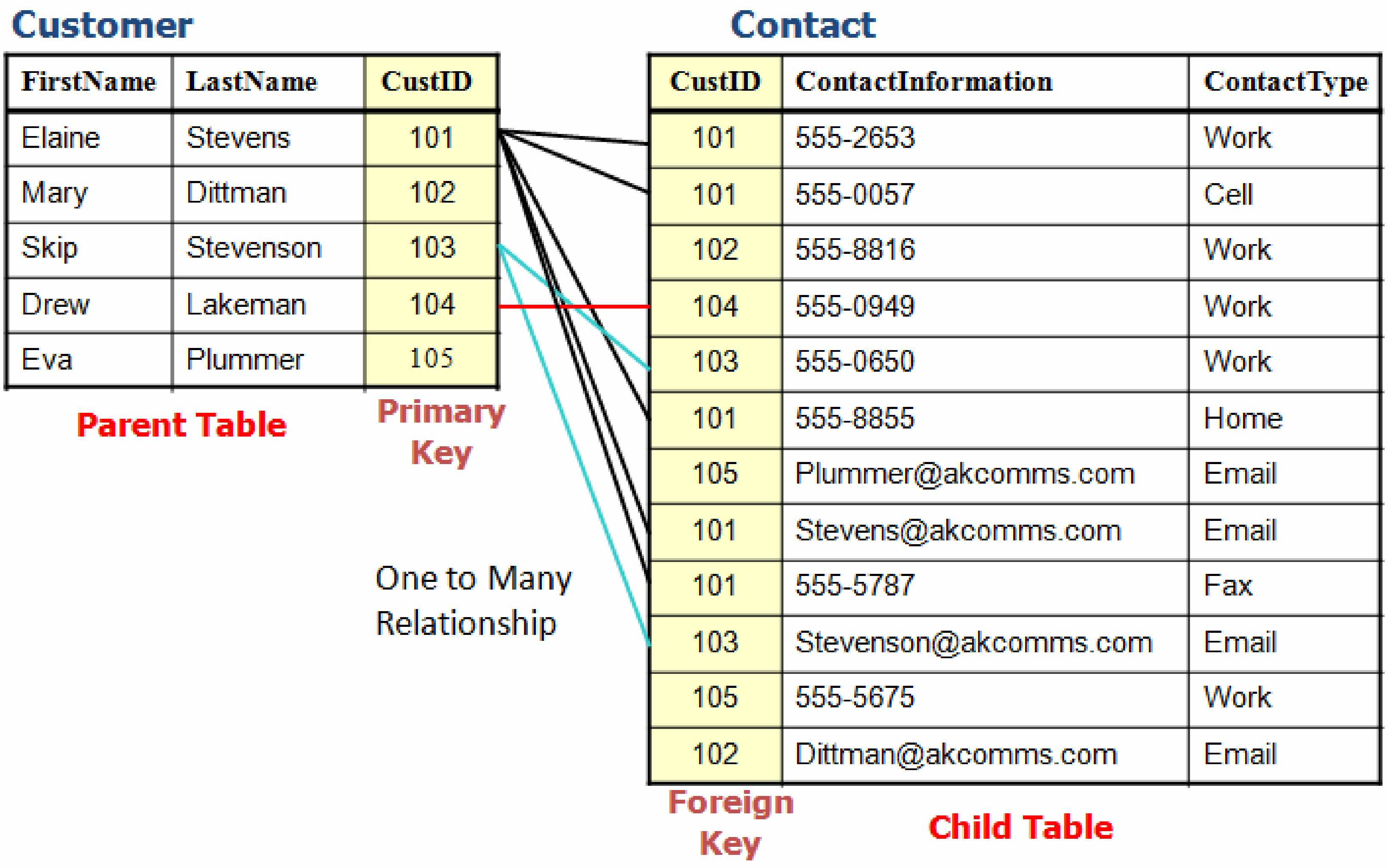
- Domain constraints
  - Given by dom(A_i)
- Key constraints
- Null constraints
  - (which attribute can/cannot be null)
- Entity-integrity constraints
- Referential integrity constraints

Referential integrity constraint

- Constraints that hold for multiple relation schemas at once, meant to guard the consistency among relations
- *Informal*: a tuple that refers to a tuple in another relation must always refer to an existing tuple
- *Formally*: foreign key

Foreign key

- A set attributes FK of a relation schema R_1 is a *foreign key* of R_1 *referring to* R_2 if
  - a 1-to-1 relation exists between attributes a_i in FK and attributes a_j in the primary key PK of R_2 such that $dom(a_i) = dom(a_j)$;
  - the value of FK in all $t_1 \in r(R_1)$ either is *null* or exists as value of PK in $t_2 \in r(R_2)$:
    $\forall\ t_1 \in r(R_1)$: { $t_1[FK] = null \vee$
    $\exists\ t_2 \in r(R_2)$: $t_1[FK] = t_2[PK]$ }.



Customer

FirstName	LastName	CustID
Elaine	Stevens	101
Mary	Dittman	102
Skip	Stevenson	103
Drew	Lakeman	104
Eva	Plummer	105

Parent Table

Primary Key

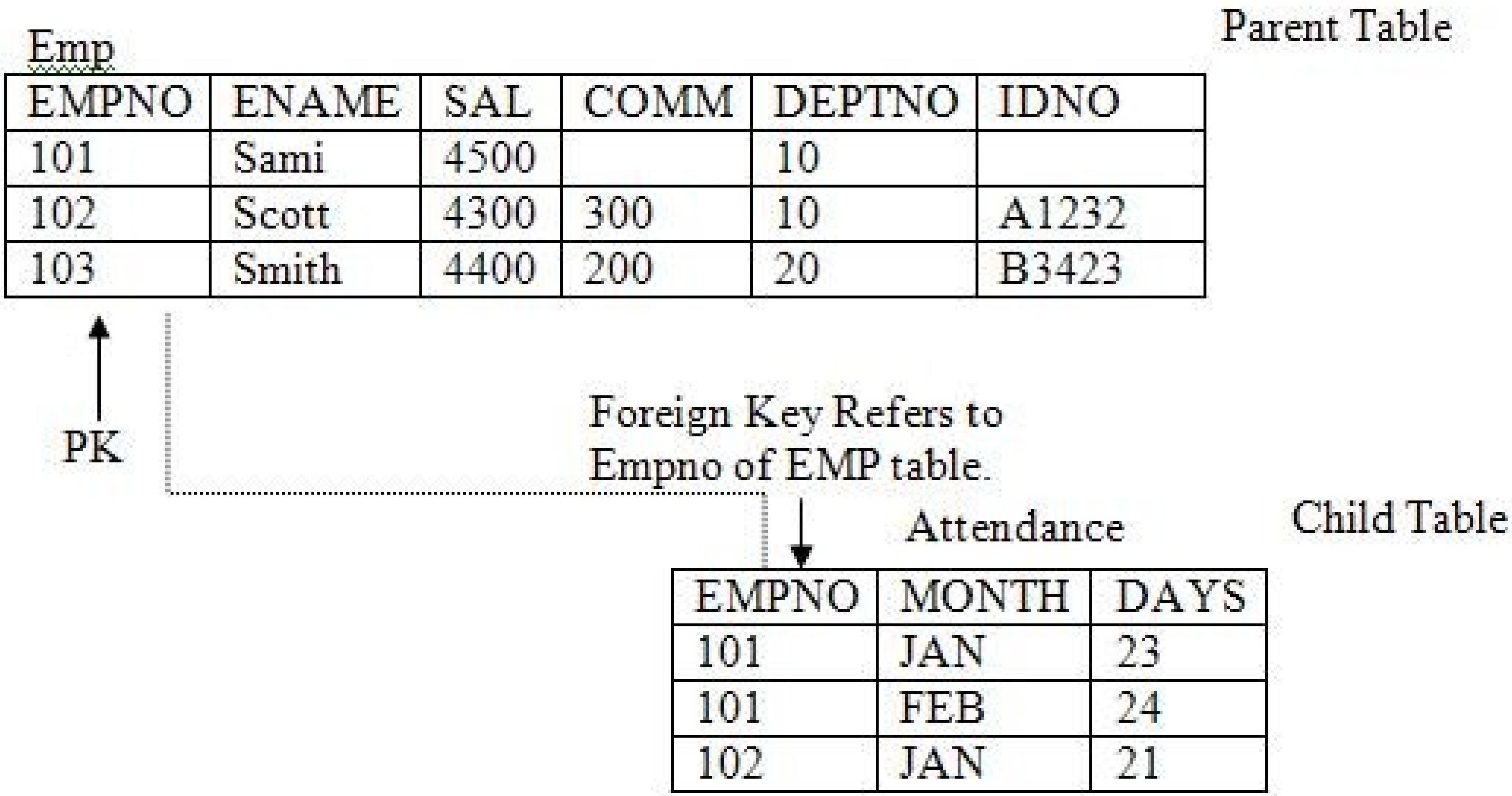
Contact

CustID	ContactInformation	ContactType
101	555-2653	Work
101	555-0057	Cell
102	555-8816	Work
104	555-0949	Work
103	555-0650	Work
101	555-8855	Home
105	Plummer@akcomms.com	Email
101	Stevens@akcomms.com	Email
101	555-5787	Fax
103	Stevenson@akcomms.com	Email
105	555-5675	Work
102	Dittman@akcomms.com	Email

One to Many Relationship

Foreign Key

Child Table



Emp

Parent Table

EMPNO	ENAME	SAL	COMM	DEPTNO	IDNO
101	Sami	4500		10	
102	Scott	4300	300	10	A1232
103	Smith	4400	200	20	B3423

PK

Foreign Key Refers to
Empno of EMP table.

Attendance

Child Table

EMPNO	MONTH	DAYS
101	JAN	23
101	FEB	24
102	JAN	21

Integrity constraints

- These five restrictions belong to a relational database schema
- Some of them are indicated graphically: *primary key* and *foreign key*
- The DDL of a DBMS contains special constructions for integrity constraints

Other constraints (3rd category)

- The five constraints mentioned do not include *semantic* constraints
    - the salary of an employee should not exceed that of his supervisor
- To this end, special languages exist: *constraint specification language*, or mechanisms like *triggers* or *assertions* are used

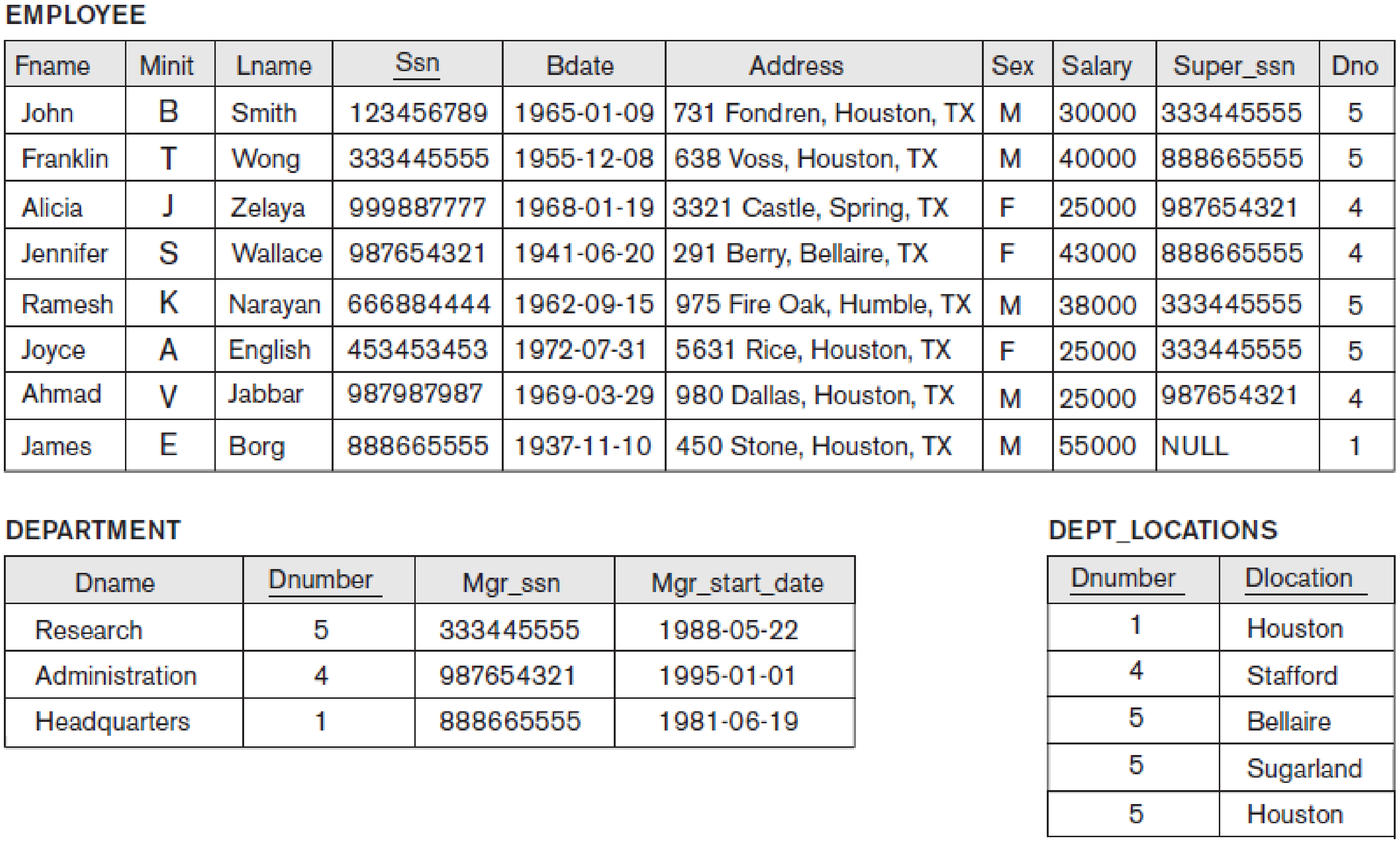
Constraint violations

- During changes of a database, the integrity constraints should not be violated
- There are three kinds of changes
  - inserting tuples to a relation
  - deleting tuples from a relation
  - updating tuples in a relation

Inserting tuples

- A new tuple t must have
  - domain values (or null) for all attributes $t[A] \in \text{dom}(A) \cup \{\text{null}\}$
  - no null for indicated attributes (when null is not allowed) and for all attributes in the primary key
  - unique values for the primary key and for all other candidate keys
  - valid foreign keys

Figure 3.6
One possible database state for the COMPANY relational database schema.



EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

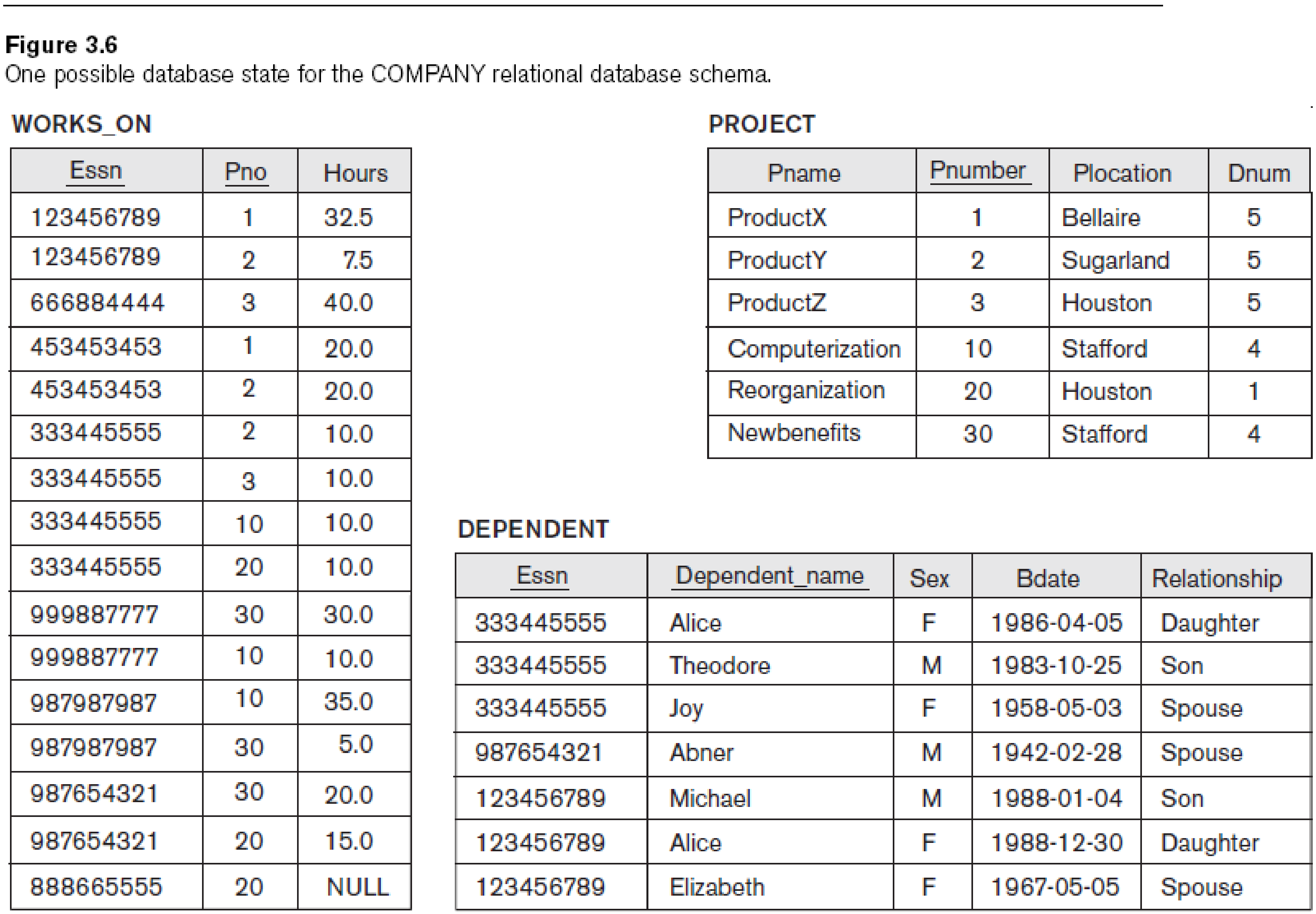


Figure 3.6
One possible database state for the COMPANY relational database schema.

WORKS_ON

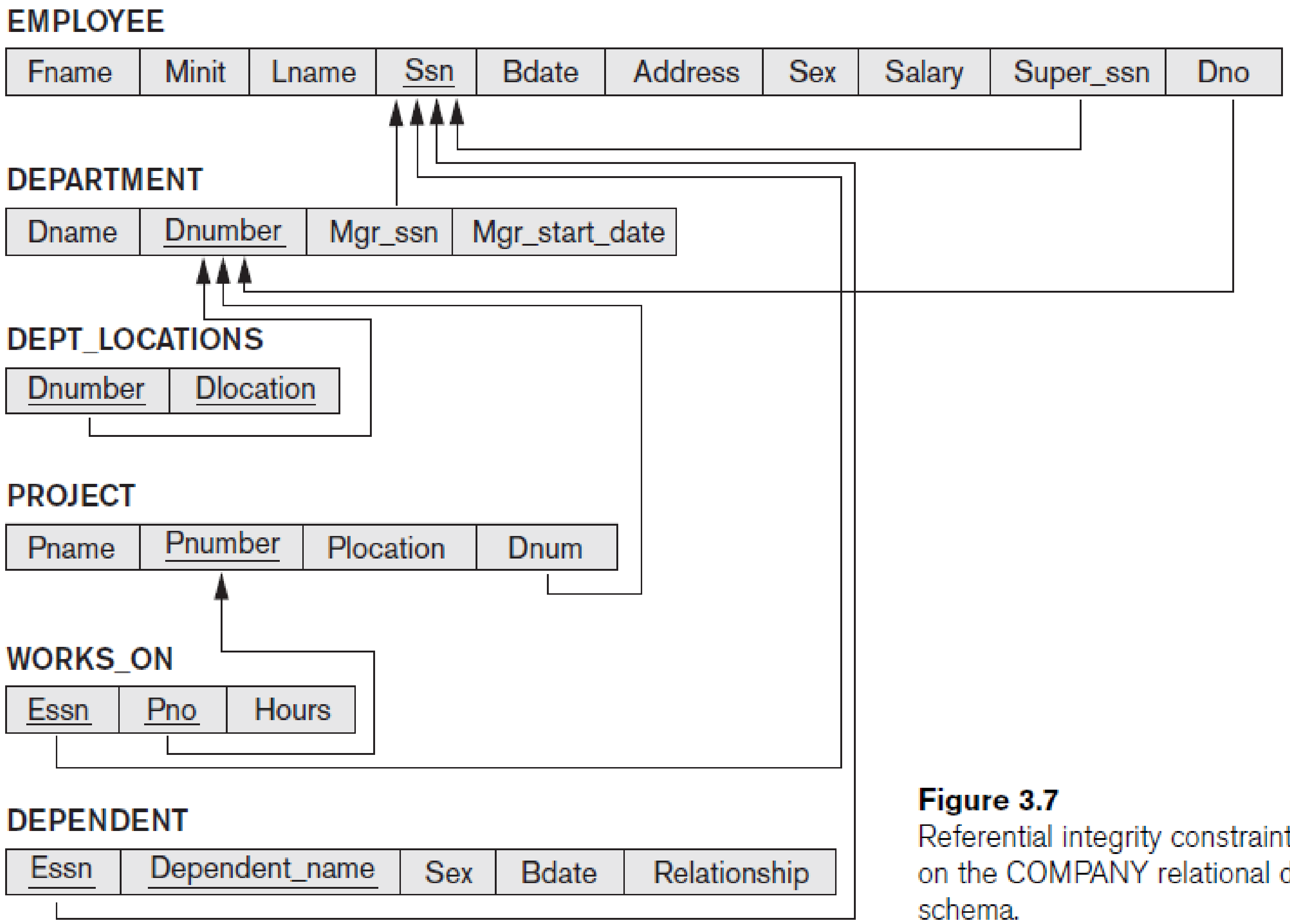
Essn	Pno	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse



EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

DEPT_LOCATIONS

Dnumber	Dlocation

PROJECT

Pname	Pnumber	Plocation	Dnum

WORKS_ON

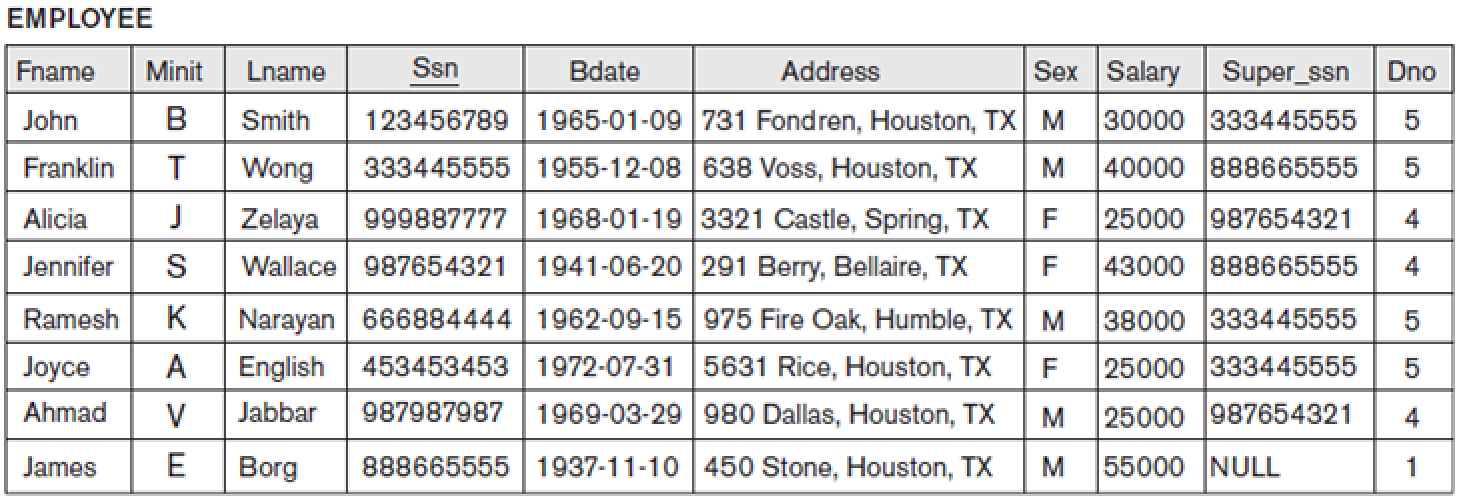
Essn	Pno	Hours

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship

Figure 3.7
Referential integrity constraints displayed on the COMPANY relational database schema.

- *Operation*:
Insert <‘Alicia’, ‘J’, ‘Zelaya’, ‘999887777’, ‘1960-04-05’, ‘6357 Windy Lane, Katy, TX’, F, 28000, ‘987654321’, 4> into EMPLOYEE.
Result: This insertion violates the key constraint because another tuple with the same Ssn value already exists in the EMPLOYEE relation, and so it is rejected.
- *Operation*:
Insert <‘Cecilia’, ‘F’, ‘Kolonsky’, ‘677678989’, ‘1960-04-05’, ‘6357 Windswept, Katy, TX’, F, 28000, ‘987654321’, 7> into EMPLOYEE.
Result: This insertion violates the referential integrity constraint specified on Dno in EMPLOYEE because no corresponding referenced tuple exists in DEPARTMENT with Dnumber = 7.



EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

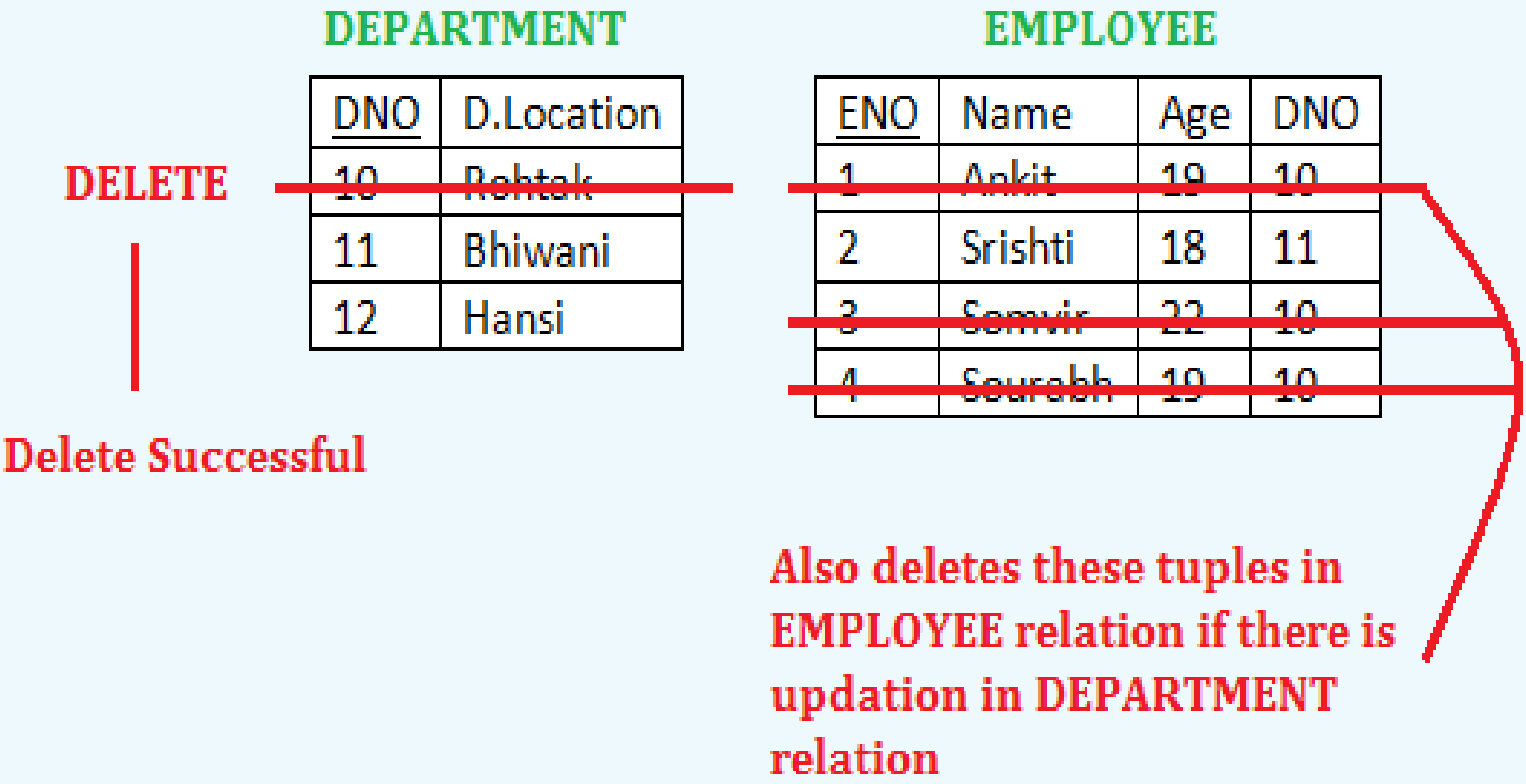
Deleting tuples

- No reference to this tuple (t_1) must exist: $\neg \exists\, t_2 \in r(R_2)\colon\; t_2[FK] = t_1[PK]$
- Three reactions when t_1 is referred to:
  - refuse deletion;
  - propagate (cascade): also delete t_2 etc.
  - change t_2, e.g. make $t_2[FK]$ = null.
- This reaction is defined with the foreign key

- *Operation*:
  Delete the EMPLOYEE tuple with Ssn = '333445555'.
  Result: This deletion will result in even worse referential integrity violations, because the tuple involved is referenced by tuples from the EMPLOYEE, DEPARTMENT, WORKS_ON, and DEPENDENT relations.

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1



DEPARTMENT

DNO	D.Location
~~10~~	~~Rohtak~~
11	Bhiwani
12	Hansi

EMPLOYEE

ENO	Name	Age	DNO
~~1~~	~~Ankit~~	~~19~~	~~10~~
2	Srishti	18	11
~~3~~	~~Somvir~~	~~22~~	~~10~~
~~4~~	~~Sourabh~~	~~19~~	~~10~~

DELETE

Delete Successful

Also deletes these tuples in EMPLOYEE relation if there is updation in DEPARTMENT relation

ON DELETE SETS NULL

DELETE

Delete Successful

DEPARTMENT

DNO	D.Location
~~10~~	~~Rohtak~~
11	Bhiwani
12	Hansi

EMPLOYEE

ENO	Name	Age	DNO
~~1~~	~~Ankit~~	~~19~~	~~10~~
2	Srishti	18	11
~~3~~	~~Somvir~~	~~22~~	~~10~~
~~4~~	~~Sourabh~~	~~19~~	~~10~~

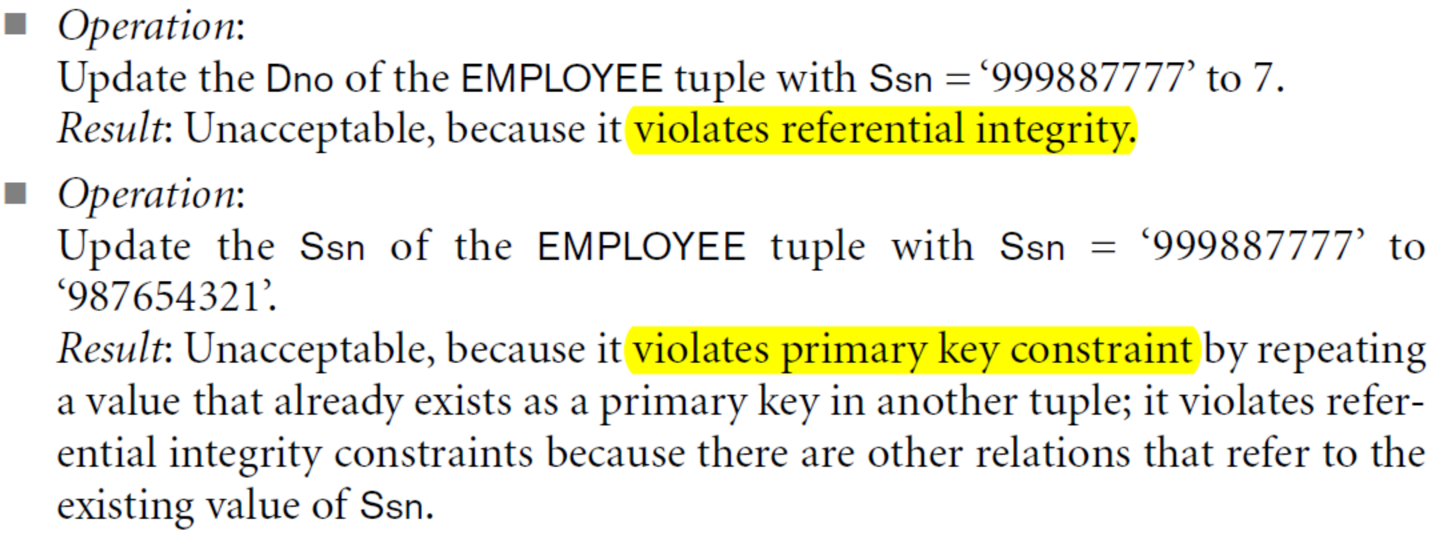
⇩

ENO	Name	Age	DNO
NULL	NULL	NULL	NULL
2	Srishti	18	11
NULL	NULL	NULL	NULL
NULL	NULL	NULL	NULL

Puts NULL in referencing relation if there is deletion in referenced relation

Update tuples

- After update the tuple must have
  - no extra-domain values or forbidden null values
  - valid foreign keys
  - unique key values
- When the primary key is changed, either:
  - refuse change,
  - change referring keys to null
  - change referring keys to the new value (propagate)



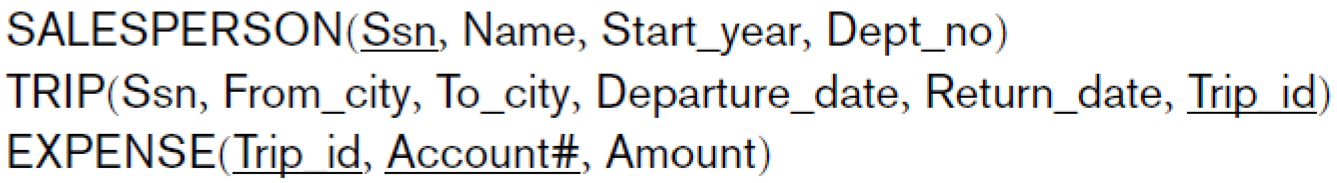
- *Operation*:
  Update the Dno of the EMPLOYEE tuple with Ssn = ‘999887777’ to 7.
  Result: Unacceptable, because it violates referential integrity.
- *Operation*:
  Update the Ssn of the EMPLOYEE tuple with Ssn = ‘999887777’ to ‘987654321’.
  Result: Unacceptable, because it violates primary key constraint by repeating a value that already exists as a primary key in another tuple; it violates referential integrity constraints because there are other relations that refer to the existing value of Ssn.

The Transaction Concept

- **Transaction**
  - Executing program
  - Includes some database operations
  - Must leave the database in a valid or consistent state
- **Online transaction processing (OLTP) systems**
  - Execute transactions at rates that reach several hundred per second

Quiz 1

- Consider the following relations for a database that keeps track of business trips of salespersons in a sales office:

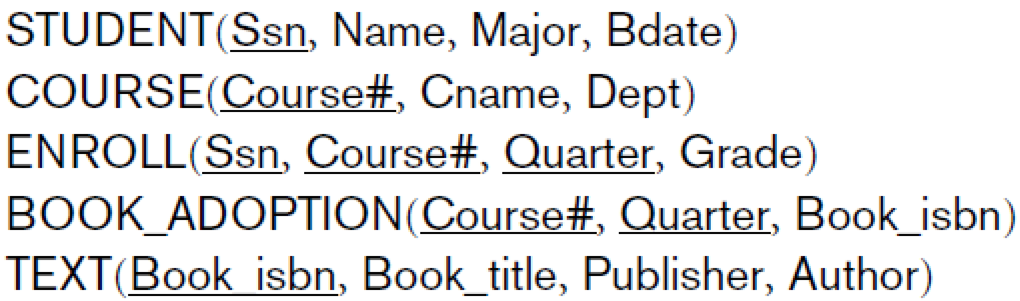


SALESPERSON(Ssn, Name, Start_year, Dept_no)
TRIP(Ssn, From_city, To_city, Departure_date, Return_date, Trip_id)
EXPENSE(Trip_id, Account#, Amount)

- A trip can be charged to one or more accounts. Specify the foreign keys for this schema, stating any assumptions you make.

Quiz 2

- Consider the following relations for a database that keeps track of student enrollment in courses and the books adopted for each course:



STUDENT(Ssn, Name, Major, Bdate)
COURSE(Course#, Cname, Dept)
ENROLL(Ssn, Course#, Quarter, Grade)
BOOK_ADOPTION(Course#, Quarter, Book_isbn)
TEXT(Book_isbn, Book_title, Publisher, Author)

- Specify the foreign keys for this schema, stating any assumptions you make.