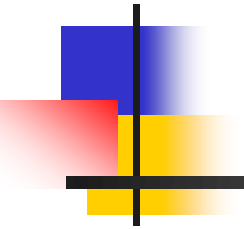
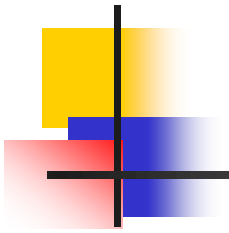


The Relational Model and Relational DB Constraints

Chapter 5

Text book sections: All

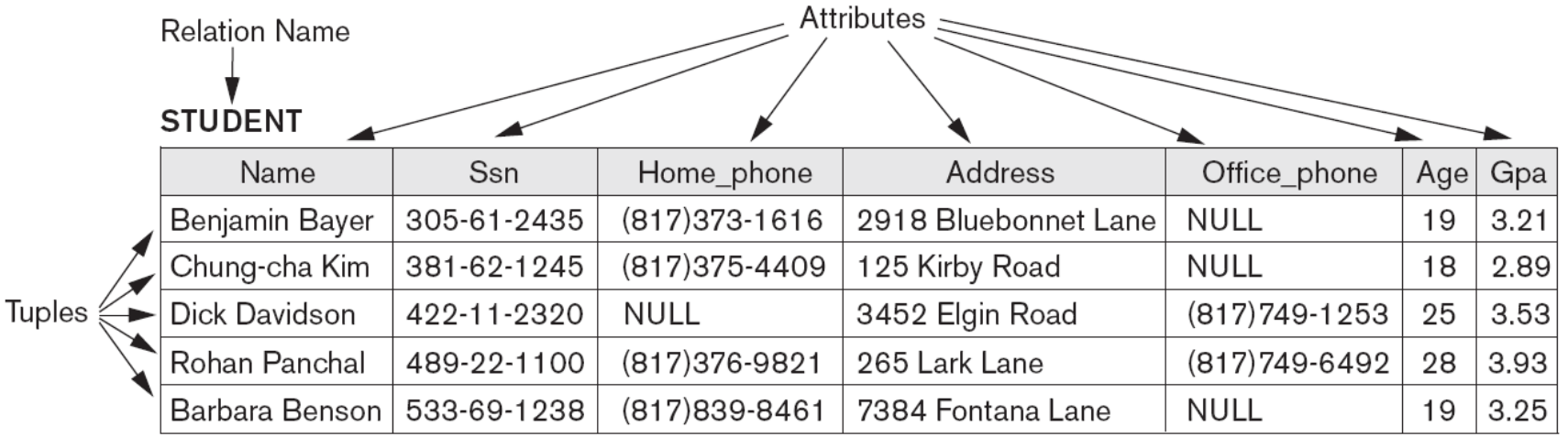


Outline

- The Relational Data Model (RDM)
- Relational Database (RDB) Constraints
- RDM Constraints and RDB Schemas
- Dealing with Constraint Violations

Relational Model Concepts

- RDM represents data as a collection of relations



Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Benjamin Bayer	305-61-2435	(817)373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	(817)375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	(817)749-1253	25	3.53
Rohan Panchal	489-22-1100	(817)376-9821	265 Lark Lane	(817)749-6492	28	3.93
Barbara Benson	533-69-1238	(817)839-8461	7384 Fontana Lane	NULL	19	3.25

Figure 3.1
The attributes and tuples of a relation STUDENT.

Domains, Attributes, Tuples, and Relations

- **Domain**: Set of atomic values specified by a **Data type**
- **Atomic:** Each value indivisible
- **Relation schema *R***
  - Denoted by $R(A_1, A_2, ..., A_n)$
  - Made up of a relation name *R* and a list of attributes, $A_1, A_2, ..., A_n$
- **Attribute A_i :** Name of a role played by some domain *D* in the relation schema *R*
- **Degree** (or **arity**) of a relation: Number of attributes in a relation
- **Cardinality:** Total number of values in domain

Characteristics of Relations

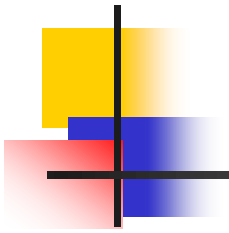
- **Relation is a set**
  - ordering of tuples in a relation is not important
  - Order of attributes is not that important
  - No two tuples can have the same combination of values for all their attributes.
- **Each value in a tuple is atomic**: Composite and multivalued attributes not allowed
- **NULL values**: Represent the values of attributes that may be unknown or may not apply to a tuple.

Relational Model Notation

- Relation schema *R* of degree *n*
  - Denoted by $R(A_1, A_2, ..., A_n)$
- Uppercase letters *Q*, *R*, *S*
  - Denote relation names
- Lowercase letters *q*, *r*, *s*
  - Denote relation states
- Letters *t*, *u*, *v*
  - Denote tuples

Relational Model Notation

- *n-tuple t* in a relation *r*(*R*)
  - Denoted by $t = <v_1, v_2, ..., v_n>$
  - v_i is the value corresponding to attribute A_i
- Component values of tuples:
  - $t[A_i]$ and $t.A_i$ refer to the value v_i in *t* for attribute A_i
  - $t[A_u, A_w, ..., A_z]$ and $t.(A_u, A_w, ..., A_z)$ refer to the subtuple of values $<v_u, v_w, ..., v_z>$ from *t* corresponding to the attributes specified in the list



Constraints

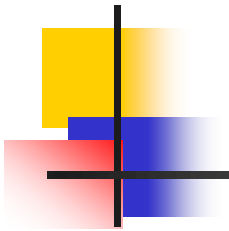
- **Constraints are:**
  - Restrictions on the actual values in a database state
  - Derived from the rules in the miniworld that the database represents
- **Why use integrity constraints?**
  - To catch data-entry errors
  - As correctness criteria when writing database updates
  - To enforce consistency across data in the database
  - To tell the system about the data

Types of Relational Model Constraints

- **Inherent model-based constraints** or **implicit constraints**
  - Inherent in the data model. Example, duplicate tuples are not allowed in a relation
- **Schema-based constraints** or **explicit constraints**
  - Can be directly expressed in schemas of the data model
- **Application-based** or **semantic constraints** or **business rules**
  - Cannot be directly expressed in schemas
  - Expressed and enforced by application program

Schema-based constraints or explicit constraints

- Domain constraint
- Key constraint
- Primary key
  - Entity integrity constraint
- Referential integrity constraint
- Not null constraint



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

Key Constraints and Constraints on NULL Values

- **Superkey:** No two distinct tuples in any state *r* of *R* can have the same value
- **Key:** A minimal superkey of R . Removing any attribute *A* from *K* leaves a set of attributes *K* that is not a superkey of *R* any more
- **Candidate key:** Relation schema may have more than one key
- **Primary key** of the relation: A designated candidate key. Other candidate keys are designated as **unique keys**

Key Constraints and Constraints on NULL Values (cont'd.)

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.

Entity, Referential Integrity, & Foreign Keys

- **Entity integrity constraint**: No primary key value can be NULL
- **Referential integrity constraint**
  - Specified between two relations
  - Maintains consistency among tuples in two relations
- **Foreign key rules:**
  - The attributes in FK have the same domain(s) as the primary key attributes PK
  - Value of FK in a tuple t_1 of the current state $r_1(R_1)$ either occurs as a value of PK for some tuple t_2 in the current state $r_2(R_2)$ or is NULL

Other Types of Constraints

- **Semantic integrity constraints**
  - May have to be specified and enforced on a relational database
  - Use **triggers** and **assertions**
  - More common to check for these types of constraints within the application programs

- **Functional dependency constraint**
  - Establishes a functional relationship among two sets of attributes *X* and *Y*
  - Value of *X* determines a unique value of *Y*

- **State constraints :** Define the constraints that a valid state of the database must satisfy

- **Transition constraints:** Define to deal with state changes in the database

Dealing with Constraint Violations

- Operations of the relational model can be categorized into retrievals and updates
- Basic operations that change the states of relations in the database:
  - Insert
  - Delete
  - Update (or Modify)

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

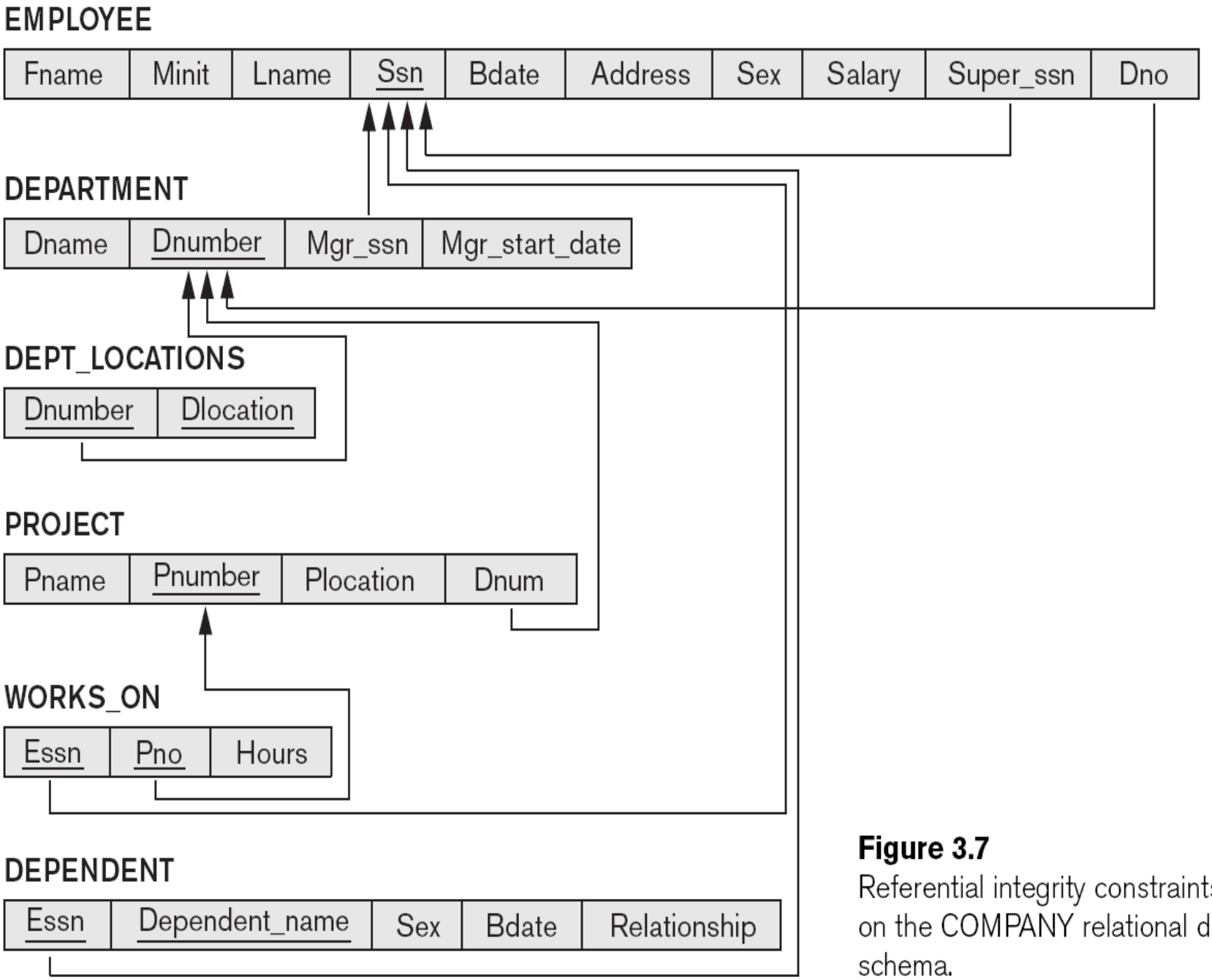
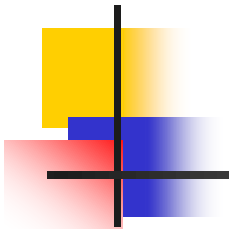
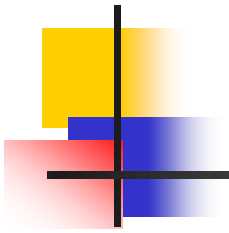


Figure 3.7
Referential integrity constraints displayed on the COMPANY relational database schema.



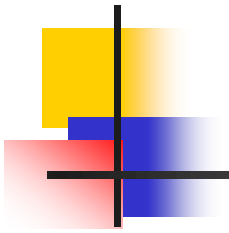
The Insert Operation

- Provides a list of attribute values for a new tuple t that is to be inserted into a relation R
- Can violate any of the four types of constraints
- If an insertion violates one or more constraints
  - Default option is to reject the insertion



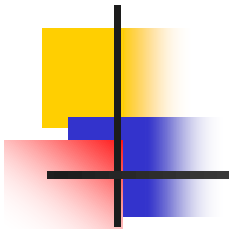
The Delete Operation

- Can violate only **referential integrity** . If tuple being deleted is referenced by foreign keys from other tuples
  - **Restrict:** Reject the deletion
  - **Cascade:** Propagate the deletion by deleting tuples that reference the tuple that is being deleted
  - **Set null** or **set default:** Modify the referencing attribute values that cause the violation



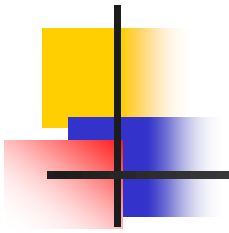
The Update Operation

- If attribute to be modified is not part of a primary key nor of a foreign key
  - Usually causes no problems
- Updating a primary/foreign key
  - Similar issues as with Insert/Delete



Summary

- Characteristics differentiate relations from ordinary tables or files
- Classify database constraints into:
  - Inherent model-based constraints
  - explicit schema-based constraints
  - application-based constraints
- Modification operations on the relational model:
  - Insert, Delete, and Update



Disclaimer

- Parts of the lecture slides contain original work from the authors of your text book

Thank You