Database Concepts (II)

Data Models

Chaokun Wang

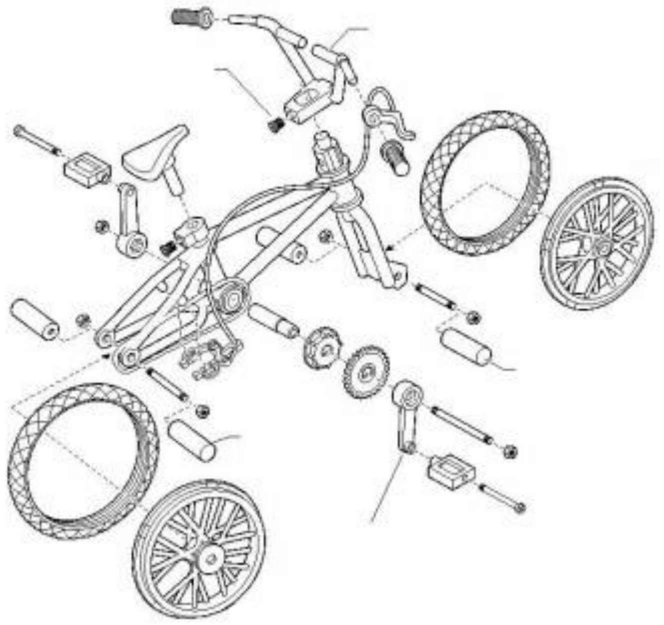
School of Software, Tsinghua University
chaokun@tsinghua.edu.cn

March 1, 2021

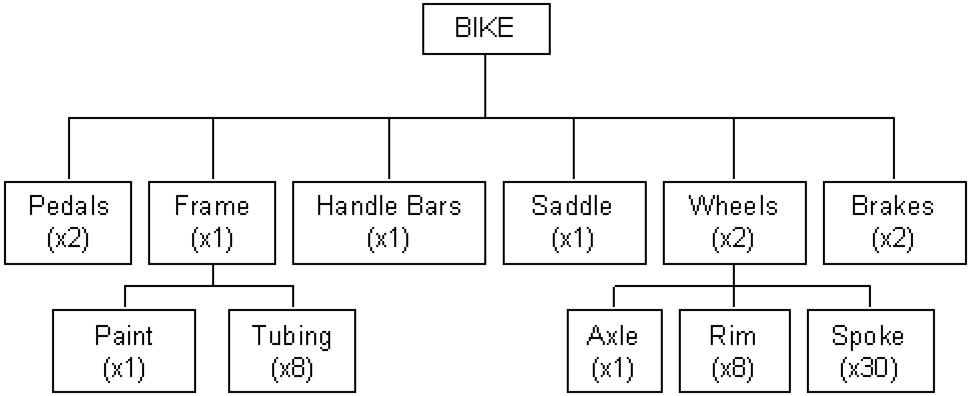
Outline

- ✈ Introduction to data models
  - Major data models
  - Data Abstraction
- Relational data model*
- No-SQL data models

How to Represent the Data?



- 按照零件拆解开？
  ->树状结构，此时就没用关系结构来存储这个bike



BIKE

Pedals (x2)

Frame (x1)

Handle Bars (x1)

Saddle (x1)

Wheels (x2)

Brakes (x2)

Paint (x1)

Tubing (x8)

Axle (x1)

Rim (x8)

Spoke (x30)

Data Model

- A relatively simple representation, usually graphical, of more complex real-world data structures
- Represents data structures and their characteristics, relations, constraints, transformations, and other constructs with the purpose of supporting a specific problem domain
  - A structure part, description of the data structure that will store the end-user data.
  - A set of integrity constraints, to guarantee the integrity of the data.
  - A manipulative part, to support the real-world data transformations.
- Data models vs. Data modeling

Basic Building Blocks of Data Models

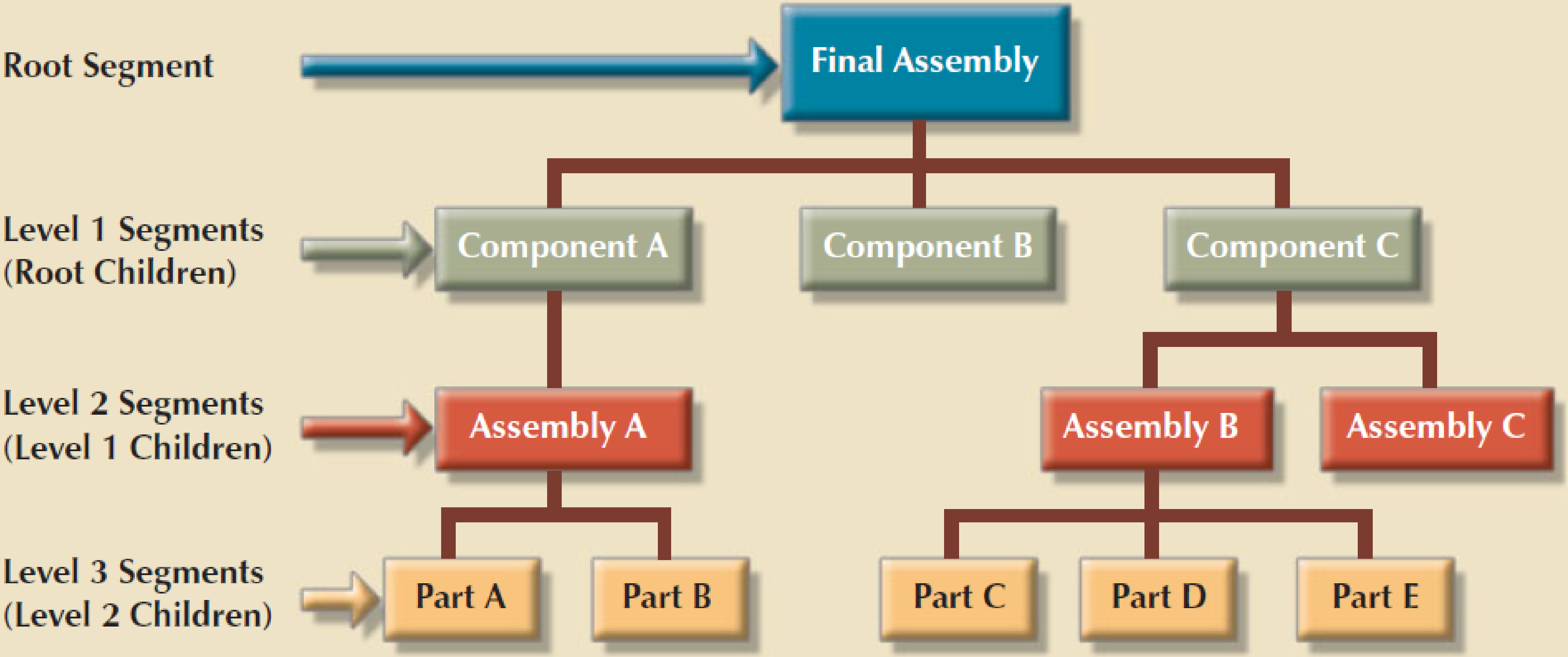
- Entity
  - Anything (a person, a place, a thing, or an event) about which data are to be collected and stored
  - A particular type of object, e.g. CUSTOMER
- Attribute
  - Characteristic of an entity, e.g. last name, address, credit limit
- Relationship
  - Association among entities
    - One-to-many (1:M or 1..*), e.g. PAINTER paints PAINTING
    - Many-to-many (M:N or *..*), e.g. STUDENT takes COURSE
    - One-to-one (1:1 or 1..1), e.g. MANAGER manages DEPARTMENT
- Constraint
  - Restriction placed on the data
    - e.g. A student's GPA must be between 0.00 and 4.00

Business Rules 业务规则

- Brief, precise, and unambiguous description of a policy, procedure, or principle within a specific organization, e.g.
  - An agent can serve many customers, and each customer can be served by only one agent
  - A training session cannot be scheduled for fewer than 10 employees or for more than 30 employees
- Translating business rules into data model components
  - Entity, relationship, constraint

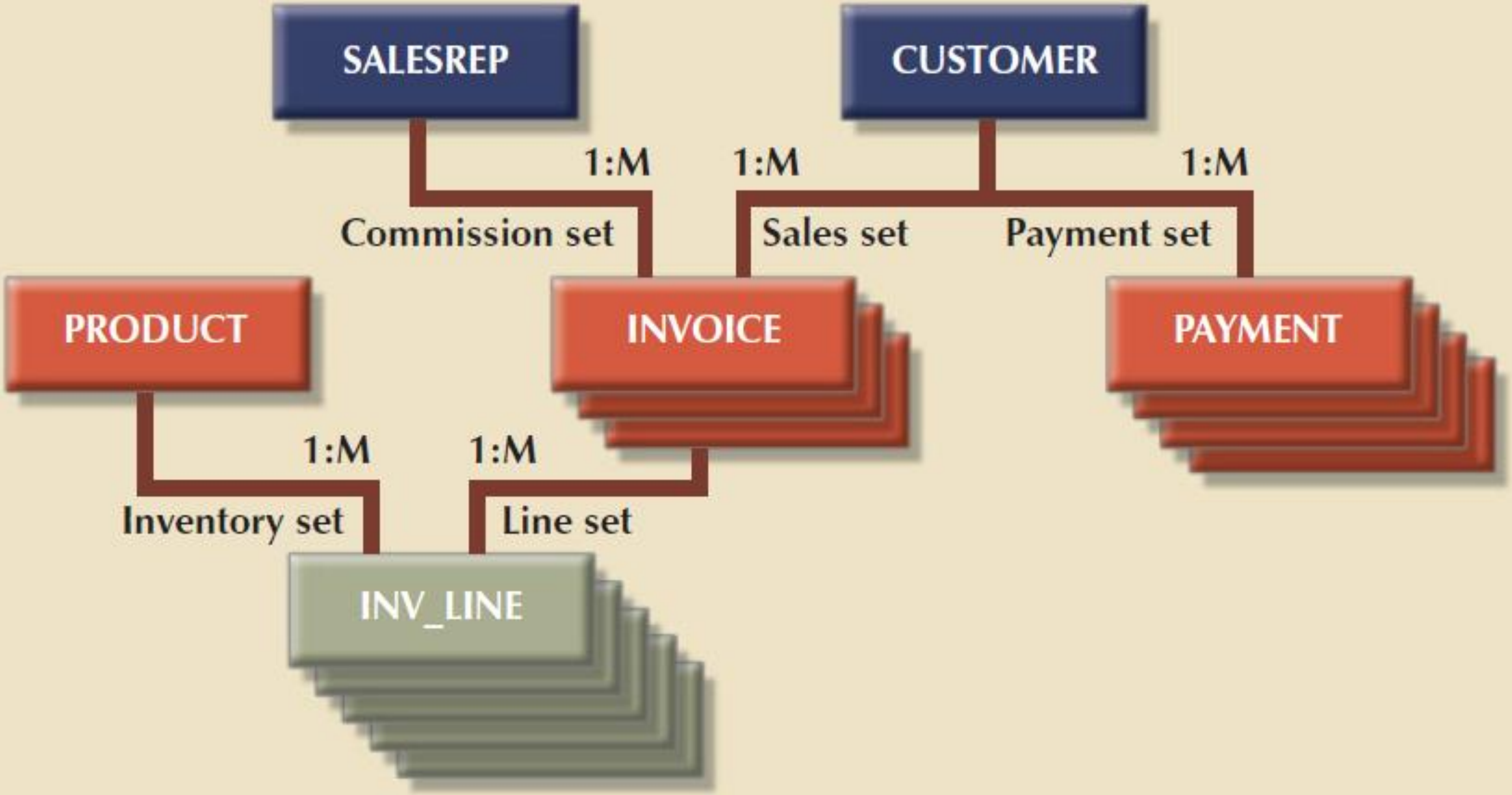
Data Models: Hierarchical Model

分层的模型

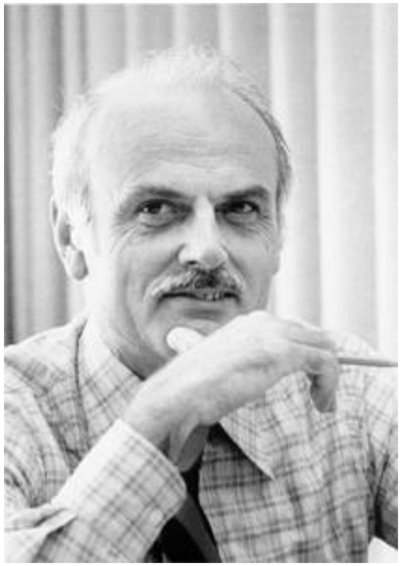


Data Models: Network Model

这种会大量用到指针



Data Models: Relational Model



Edgar Frank "Ted" Codd

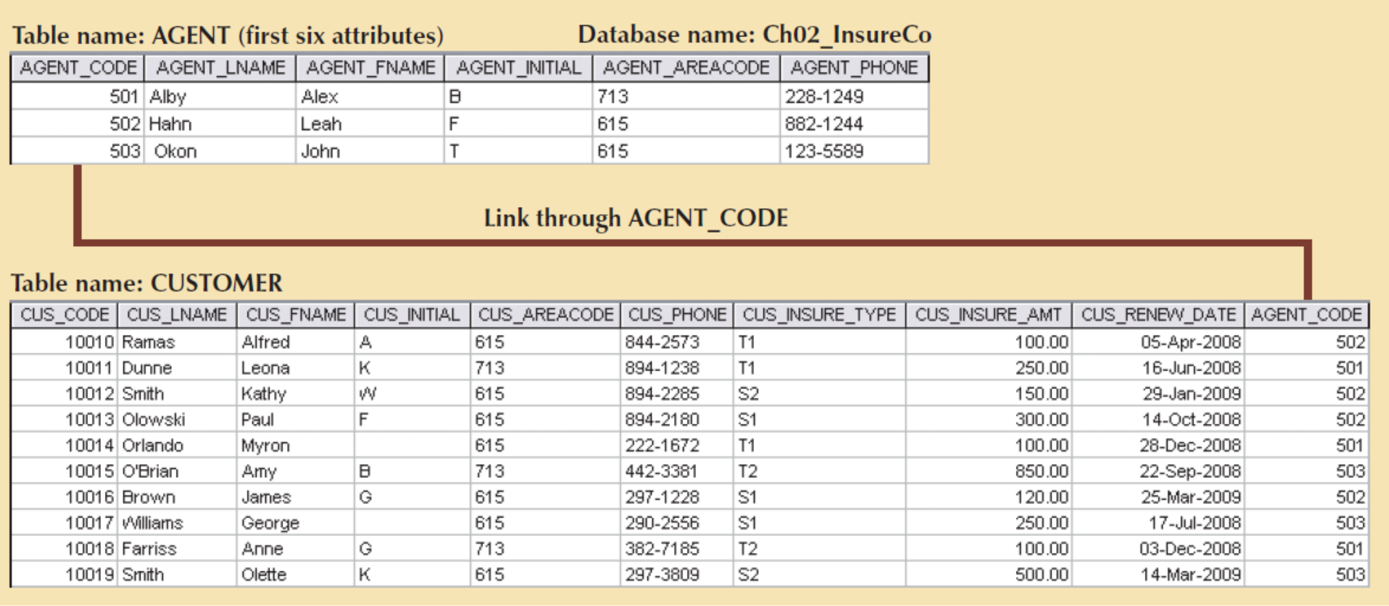


Table name: AGENT (first six attributes)

Database name: Ch02_InsureCo

AGENT_CODE	AGENT_LNAME	AGENT_FNAME	AGENT_INITIAL	AGENT_AREACODE	AGENT_PHONE
501	Alby	Alex	B	713	228-1249
502	Hahn	Leah	F	615	882-1244
503	Okon	John	T	615	123-5589

Link through AGENT_CODE

Table name: CUSTOMER

CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_AREACODE	CUS_PHONE	CUS_INSURE_TYPE	CUS_INSURE_AMT	CUS_RENEW_DATE	AGENT_CODE
10010	Ramas	Alfred	A	615	844-2573	T1	100.00	05-Apr-2008	502
10011	Dunne	Leona	K	713	894-1238	T1	250.00	16-Jun-2008	501
10012	Smith	Kathy	W	615	894-2285	S2	150.00	29-Jan-2009	502
10013	Olowski	Paul	F	615	894-2180	S1	300.00	14-Oct-2008	502
10014	Orlando	Myron		615	222-1672	T1	100.00	28-Dec-2008	501
10015	O'Brian	Amy	B	713	442-3381	T2	850.00	22-Sep-2008	503
10016	Brown	James	G	615	297-1228	S1	120.00	25-Mar-2009	502
10017	Williams	George		615	290-2556	S1	250.00	17-Jul-2008	503
10018	Farriss	Anne	G	713	382-7185	T2	100.00	03-Dec-2008	501
10019	Smith	Olette	K	615	297-3809	S2	500.00	14-Mar-2009	503

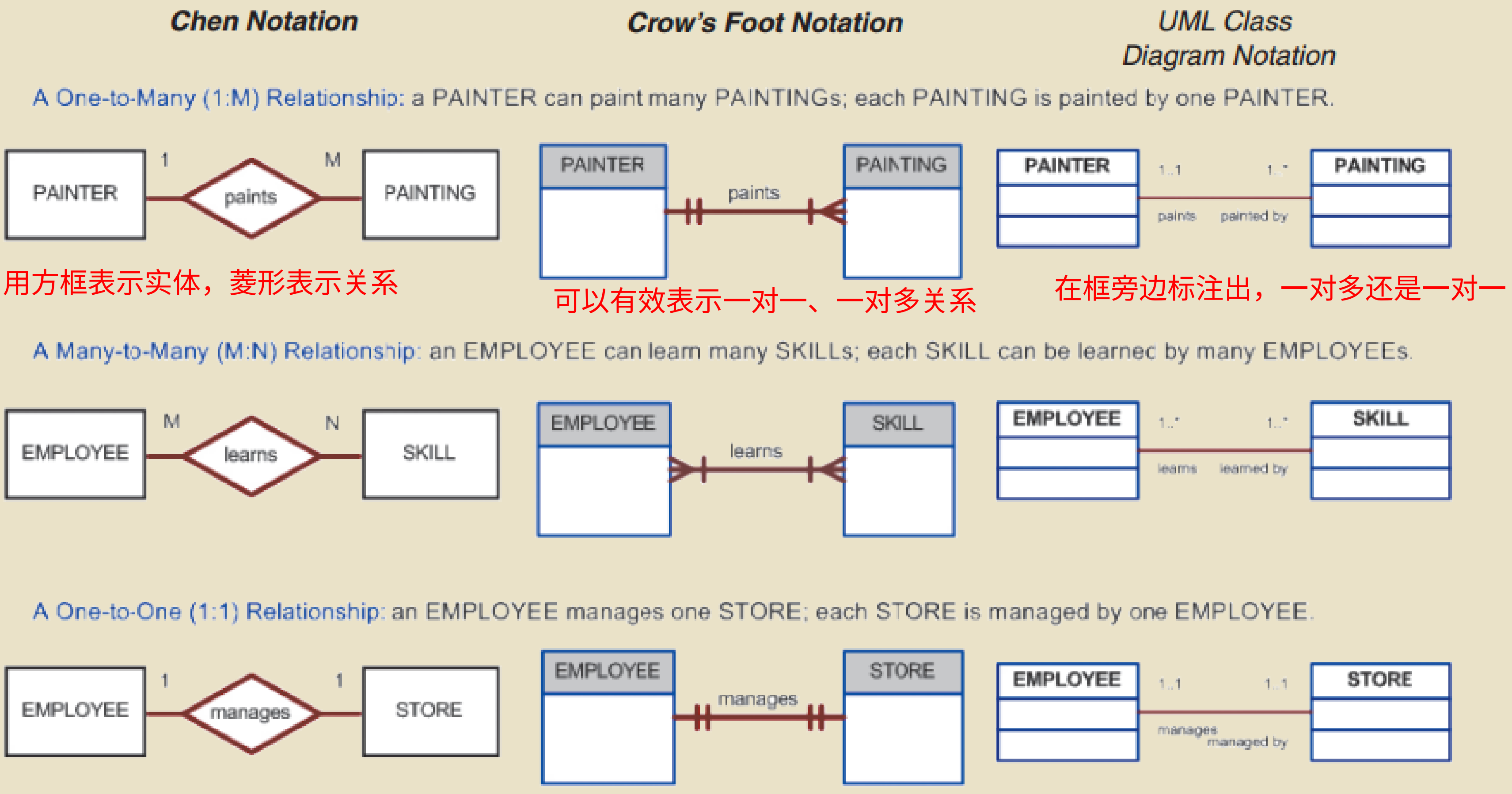
I could imagine how those queries would have been represented in CODASYL by programs that were five pages long that would navigate through this labyrinth of pointers and stuff. Codd would sort of write them down as one-liners.

—— Don Chamberlin, co-inventor of SQL

Data Models: Entity Relationship Model

这就是ER图

FIGURE 2.3 THE ER MODEL NOTATIONS



A One-to-Many (1:M) Relationship: a PAINTER can paint many PAINTINGs; each PAINTING is painted by one PAINTER.

用方框表示实体，菱形表示关系

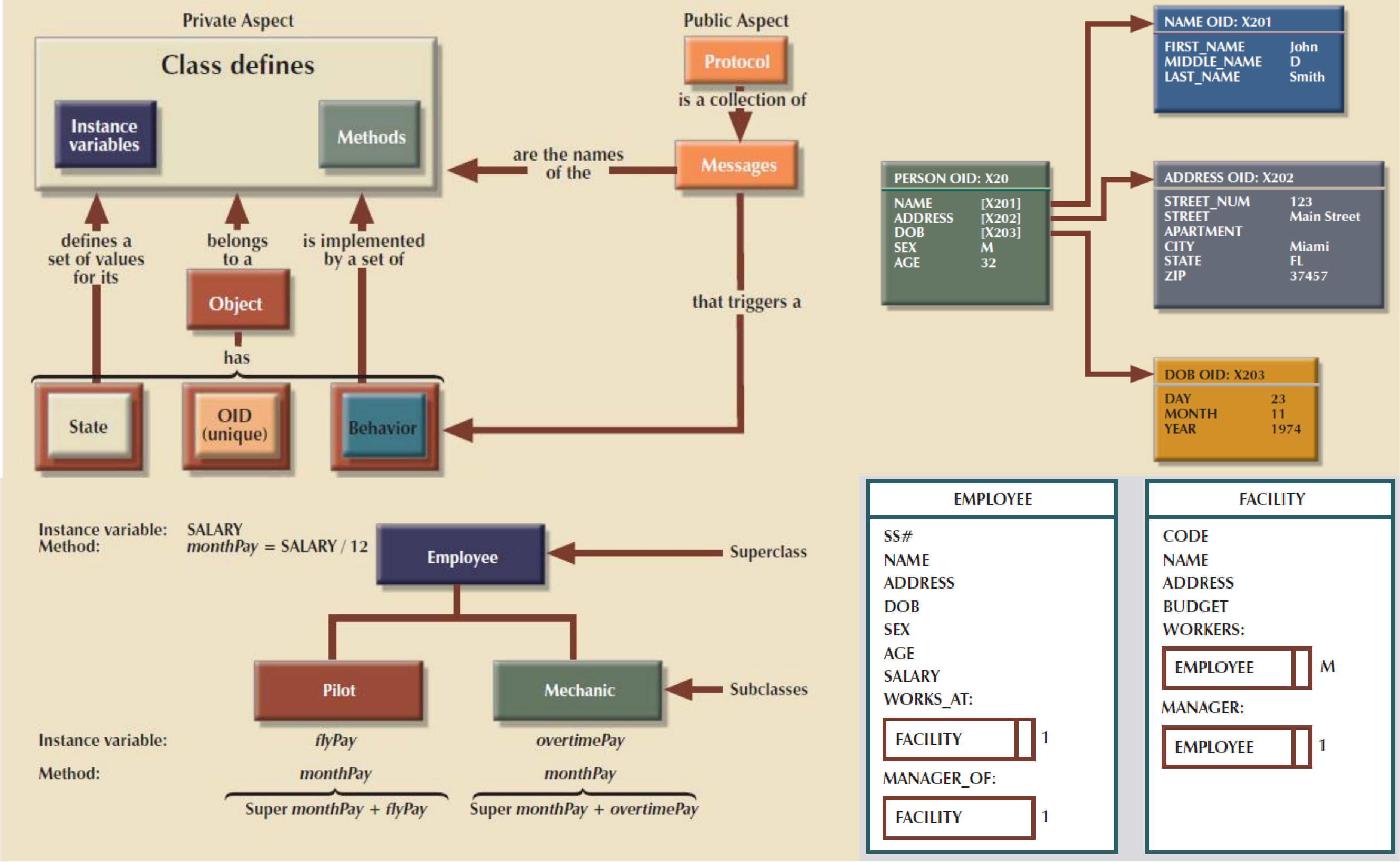
可以有效表示一对一、一对多关系

在框旁边标注出，一对多还是一对一

A Many-to-Many (M:N) Relationship: an EMPLOYEE can learn many SKILLs; each SKILL can be learnec by many EMPLOYEEs.

A One-to-One (1:1) Relationship: an EMPLOYEE manages one STORE; each STORE is managed by one EMPLOYEE.

Data Models: Object-Oriented Model



Private Aspect

Class defines

Instance variables

Methods

Public Aspect

Protocol

is a collection of

Messages

are the names of the

defines a set of values for its

belongs to a

is implemented by a set of

Object

has

that triggers a

State

OID (unique)

Behavior

NAME OID: X201

FIRST_NAME	John
MIDDLE_NAME	D
LAST_NAME	Smith

PERSON OID: X20

NAME	[X201]
ADDRESS	[X202]
DOB	[X203]
SEX	M
AGE	32

ADDRESS OID: X202

STREET_NUM	123
STREET	Main Street
APARTMENT	
CITY	Miami
STATE	FL
ZIP	37457

DOB OID: X203

DAY	23
MONTH	11
YEAR	1974

Instance variable: SALARY
Method: *monthPay* = SALARY / 12

Employee

Superclass

Pilot

Mechanic

Subclasses

Instance variable:	*flyPay*	*overtimePay*
Method:	*monthPay*	*monthPay*
	Super *monthPay* + *flyPay*	Super *monthPay* + *overtimePay*

EMPLOYEE

SS#
NAME
ADDRESS
DOB
SEX
AGE
SALARY
WORKS_AT:
FACILITY 1
MANAGER_OF:
FACILITY 1

FACILITY

CODE
NAME
ADDRESS
BUDGET
WORKERS:
EMPLOYEE M
MANAGER:
EMPLOYEE 1

Data Models: Object/Relational and XML

- Extended relational data model (ERDM)
  - Supports OO features, extensible data types based on classes, and inheritance
    - Object/relational database management system (O/R DBMS): based on ERDM
- Extensible Markup Language (XML)
  - Manages unstructured data for efficient and effective exchange of structured, semistructured, and unstructured data

Emerging Data Models: Big Data and NoSQL

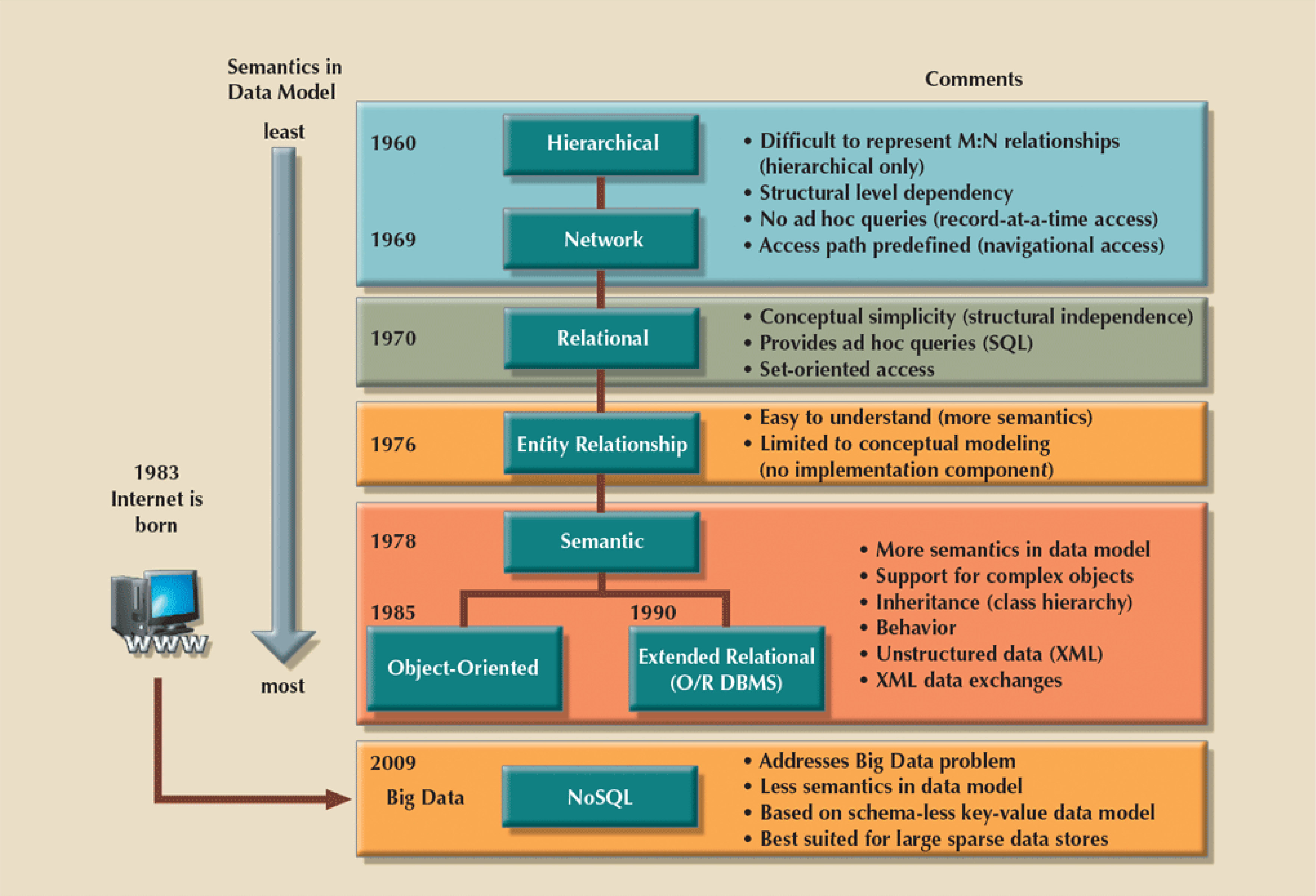
- Goals of Big Data
  - Find new and better ways to manage large amounts of web and sensor-generated data
  - Provide high performance at a reasonable cost
- Characteristics of Big Data
  - Volume 数据体量大
  - Velocity 速度快
  - Variety 数据类型多
- Challenges of Big Data
  - Volume doesn’t allow usage of conventional structures
  - Expensive
  - OLAP tools proved inconsistent dealing with unstructured data

Emerging Data Models: Big Data and NoSQL

- New technologies of Big Data
  - Hadoop
  - Hadoop Distributed File System (HDFS)
  - MapReduce
  - NoSQL
- NoSQL databases
  - Not based on the relational model
  - Support distributed database architectures
  - Provide high scalability, high availability, and fault tolerance
  - Support large amounts of sparse data
  - Geared toward performance rather than transaction consistency
  - Provides a broad umbrella for data storage and manipulation

The development of data models

FIGURE 2.5 THE EVOLUTION OF DATA MODELS



Semantics in Data Model

least

most

Comments

1960 Hierarchical

1969 Network

- Difficult to represent M:N relationships (hierarchical only)
- Structural level dependency
- No ad hoc queries (record-at-a-time access)
- Access path predefined (navigational access)

1970 Relational

- Conceptual simplicity (structural independence)
- Provides ad hoc queries (SQL)
- Set-oriented access

1976 Entity Relationship

- Easy to understand (more semantics)
- Limited to conceptual modeling (no implementation component)

1978 Semantic

1985 Object-Oriented

1990 Extended Relational (O/R DBMS)

- More semantics in data model
- Support for complex objects
- Inheritance (class hierarchy)
- Behavior
- Unstructured data (XML)
- XML data exchanges

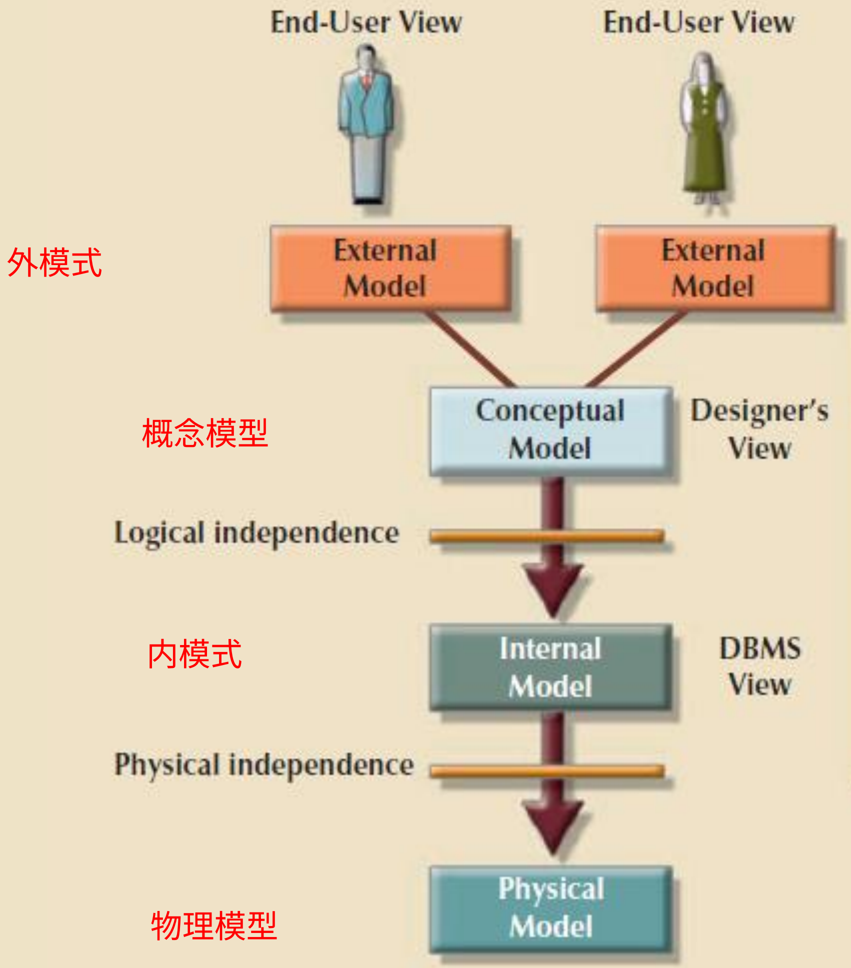
1983 Internet is born

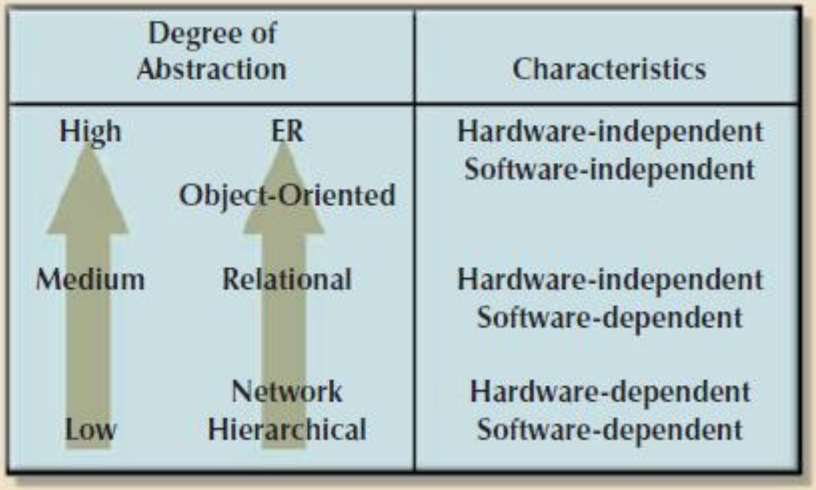
WWW

2009 Big Data NoSQL

- Addresses Big Data problem
- Less semantics in data model
- Based on schema-less key-value data model
- Best suited for large sparse data stores

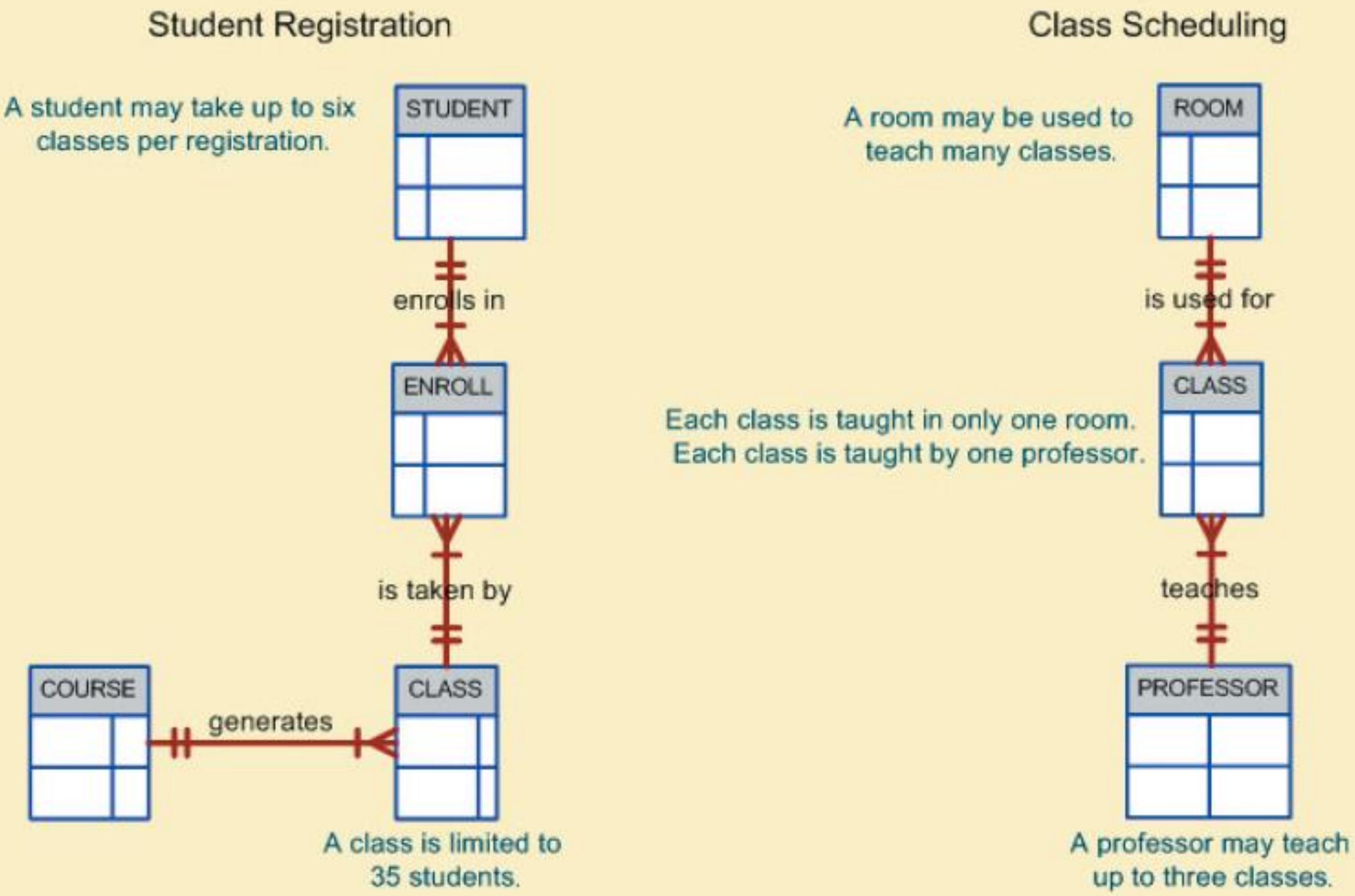
Degrees of Data Abstraction





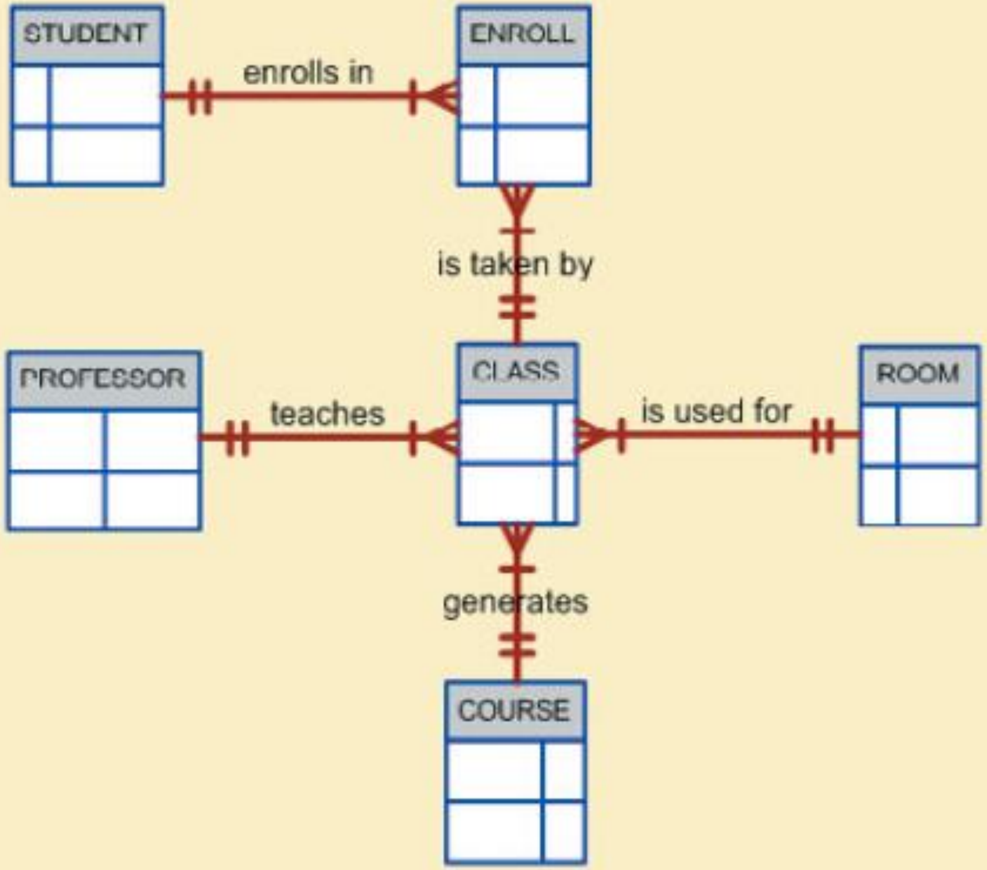
Degree of Abstraction		Characteristics
High	ER	Hardware-independent Software-independent
	Object-Oriented	
Medium	Relational	Hardware-independent Software-dependent
Low	Network Hierarchical	Hardware-dependent Software-dependent

The External Model


Student Registration
Class Scheduling
A student may take up to six classes per registration.
STUDENT
enrolls in
ENROLL
is taken by
COURSE
generates
CLASS
A class is limited to 35 students.
A room may be used to teach many classes.
ROOM
is used for
CLASS
Each class is taught in only one room.
Each class is taught by one professor.
teaches
PROFESSOR
A professor may teach up to three classes.

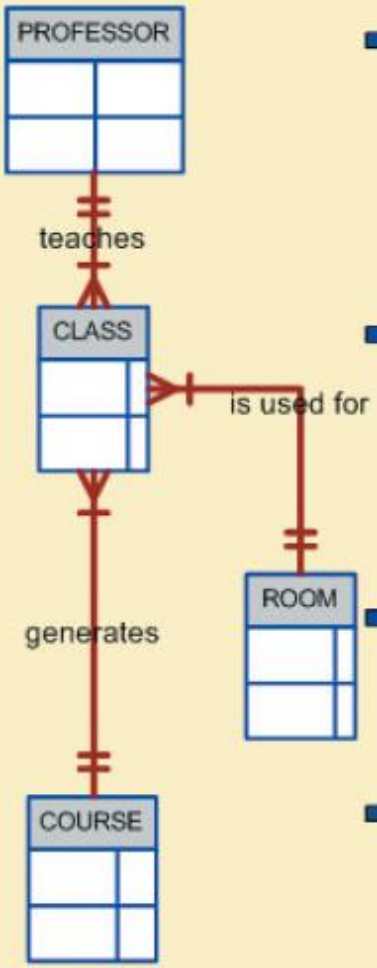
The Conceptual Model

把上一张ppt的两张图融合



The Internal Model

CONCEPTUAL MODEL



INTERNAL MODEL 表现为sql语句

```
Create Table PROFESSOR(
 PROF_ID        NUMBER PRIMARY KEY,
 PROF_LNAME     CHAR(15),
 PROF_INITIAL   CHAR(1),
 PROF_FNAME     CHAR(15),
 ..........);
```

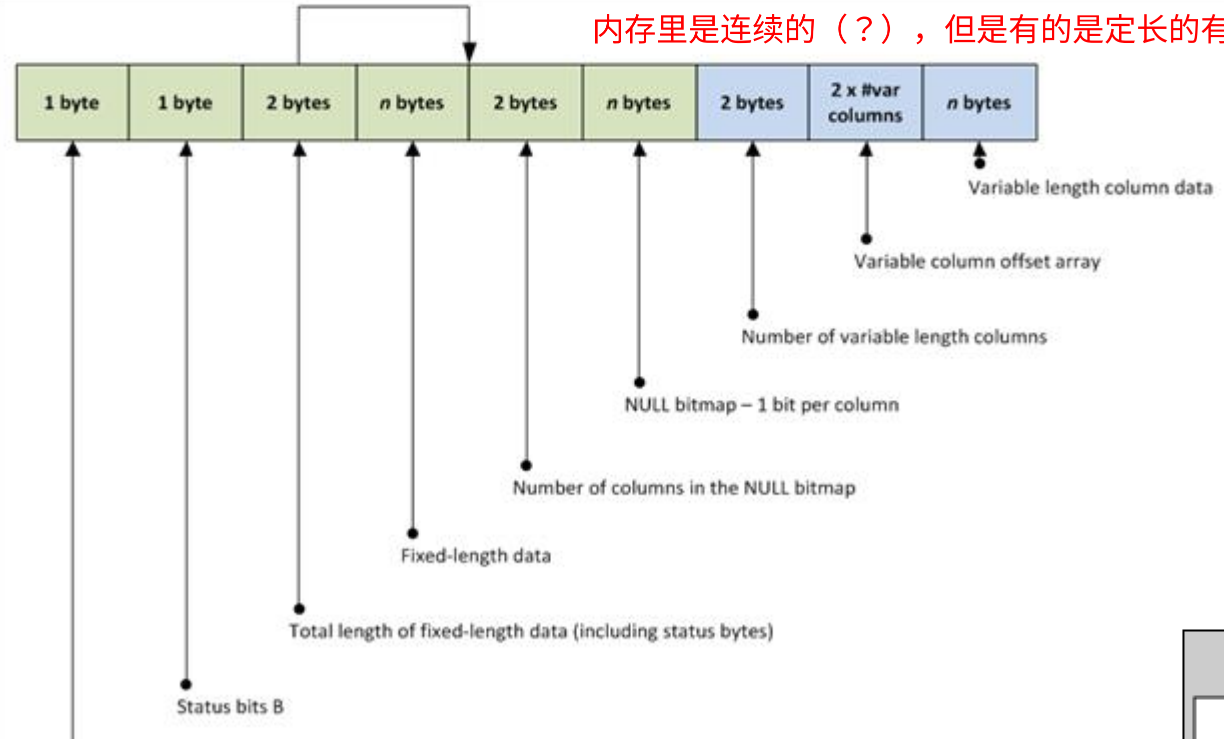
```
Create Table CLASS(
 CLASS_ID       NUMBER PRIMARY KEY,
 CRS_ID         CHAR(8)  REFERENCES COURSE,
 PROF_ID        NUMBER REFERENCES PROFESSOR,
 ROOM_ID        CHAR(8)  REFERENCES ROOM,
 ..........);
```

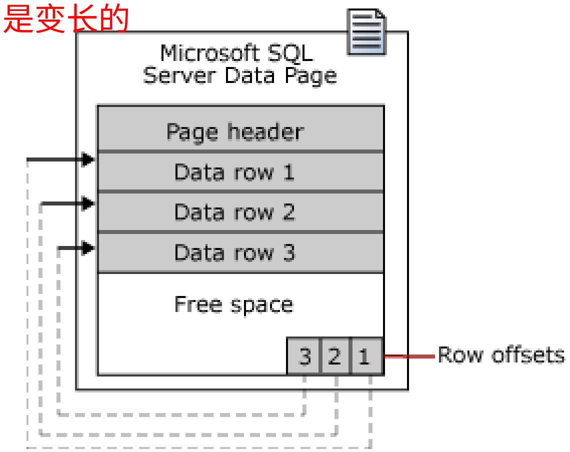
```
Create Table ROOM(
 ROOM_ID        CHAR(8) PRIMARY KEY,
 ROOM_TYPE      CHAR(3),
 ..........);
```

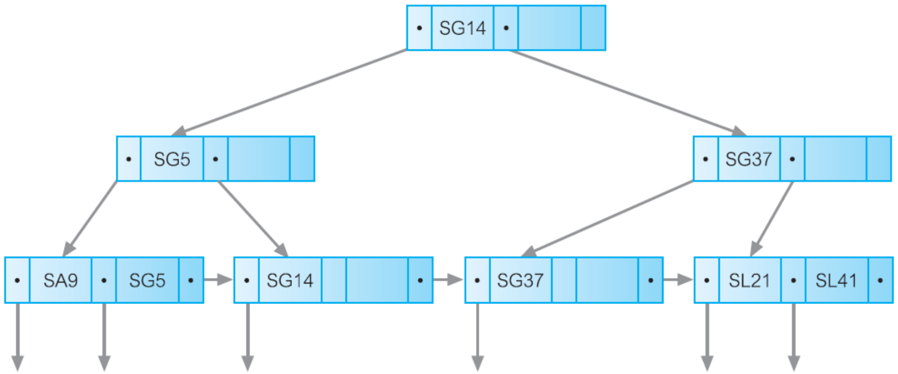
```
Create Table COURSE(
 CRS_ID         CHAR(8) PRIMARY KEY,
 CRS_NAME       CHAR(25),
 CRS_CREDITS    NUMBER,
 ..........);
```

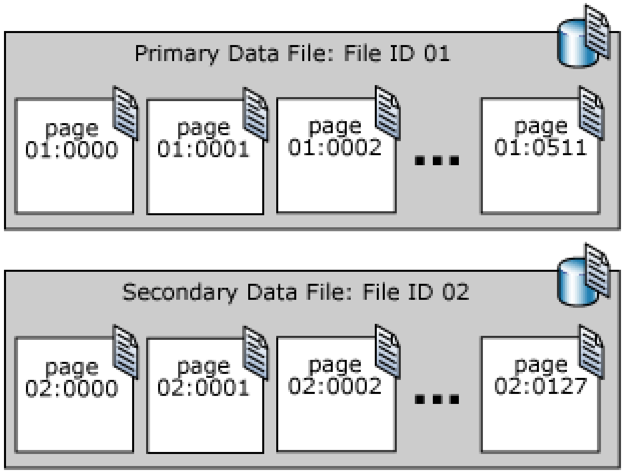
The Physical Model

内存里是连续的（？），但是有的是定长的有的是变长的


1 byte
1 byte
2 bytes
n bytes
2 bytes
n bytes
2 bytes
2 x #var columns
n bytes
Variable length column data
Variable column offset array
Number of variable length columns
NULL bitmap – 1 bit per column
Number of columns in the NULL bitmap
Fixed-length data
Total length of fixed-length data (including status bytes)
Status bits B
Status bits A





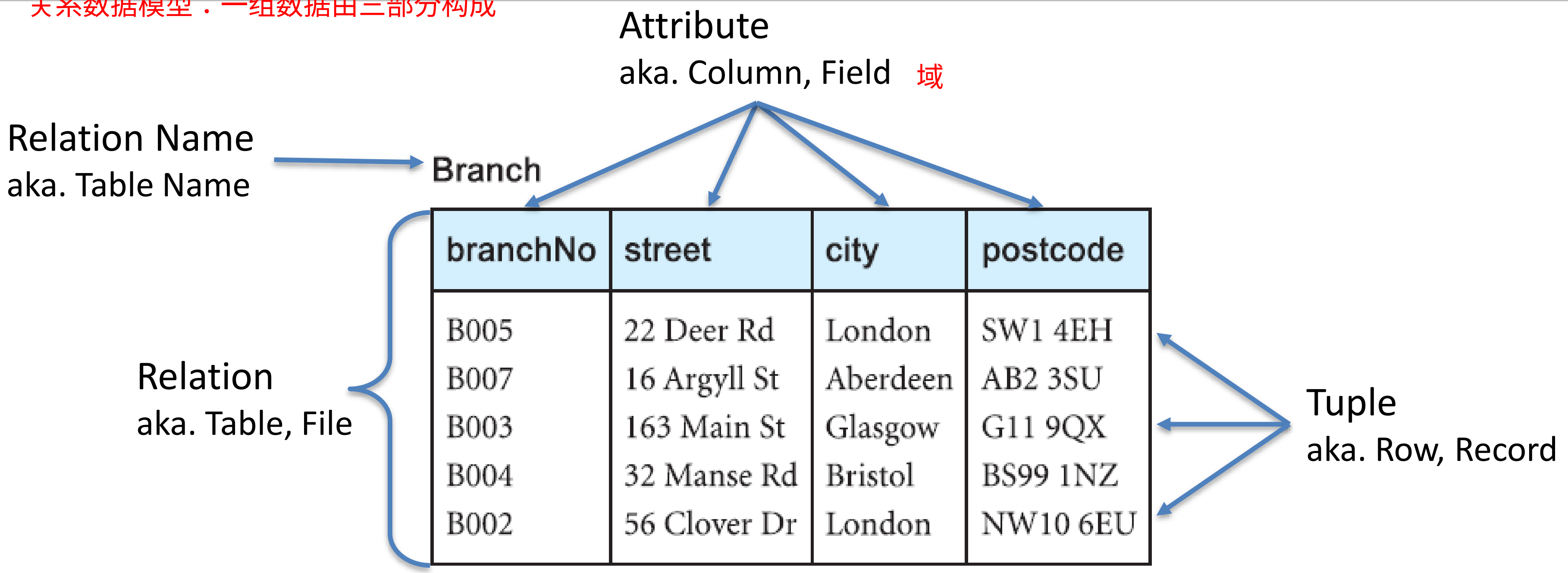


Outline

- Introduction to data models
- ✈ Relational data model
- No-SQL data models

The Relational Model

关系数据模型：一组数据由三部分构成



Attribute
aka. Column, Field 域

Relation Name
aka. Table Name

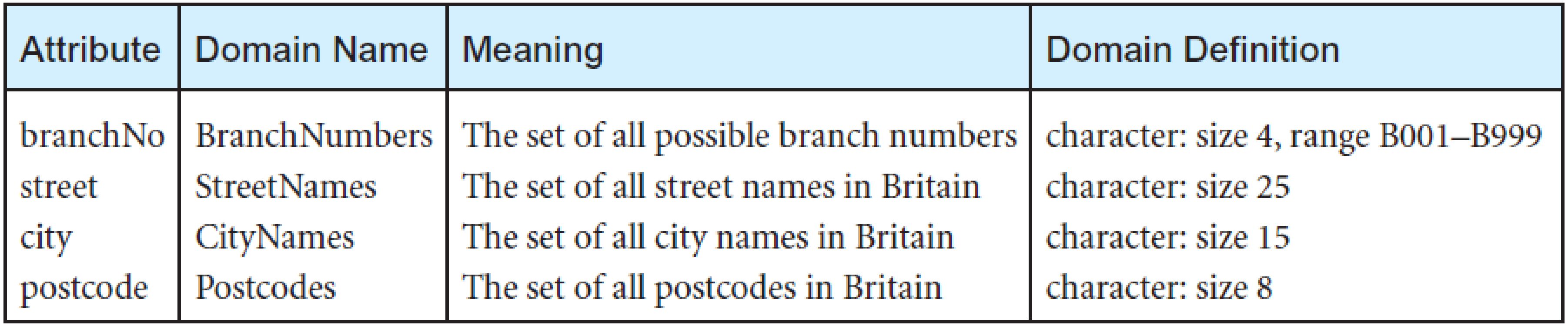
Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

Relation
aka. Table, File

Tuple
aka. Row, Record

Domain: the set of allowable values for one or more attributes

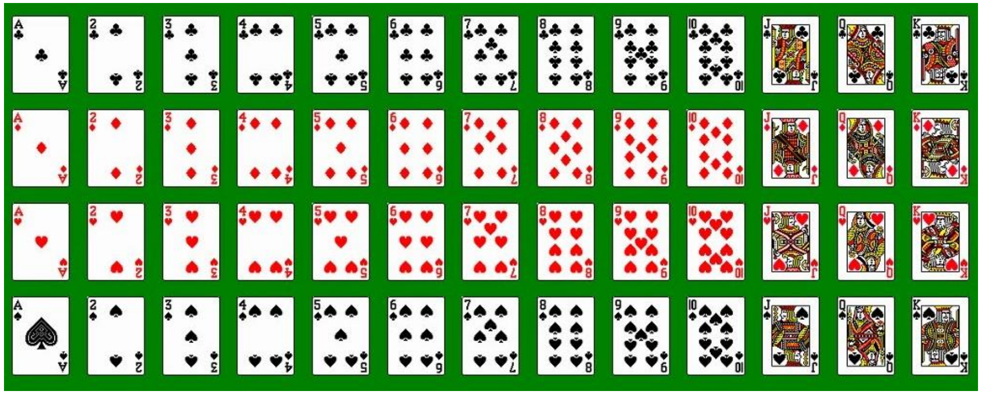


Attribute	Domain Name	Meaning	Domain Definition
branchNo	BranchNumbers	The set of all possible branch numbers	character: size 4, range B001–B999
street	StreetNames	The set of all street names in Britain	character: size 25
city	CityNames	The set of all city names in Britain	character: size 15
postcode	Postcodes	The set of all postcodes in Britain	character: size 8

Mathematical Relations

Suits: {♣, ♦, ♥, ♠}

Ranks: {A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K}



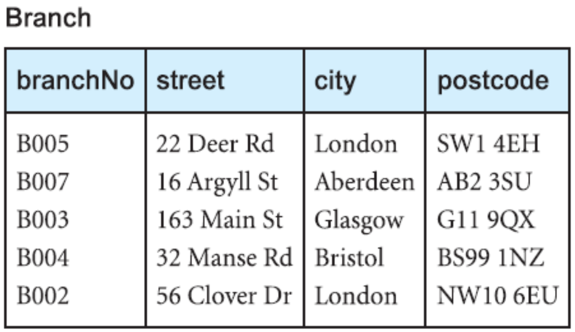
Cartesian product *Suits* × *Ranks* :

{(♣, A), (♣, 2), (♣, 3), (♣, 4), (♣, 5), ..., (♠, 9), (♠, 10), (♠, J), (♠, Q), (♠, K)}

Relation: any subset of this Cartesian product

Database Relations

- Relation schema 关系模式：构成关系的属性
  - – A named relation defined by a set of attribute and domain name pairs.
    - Let $A_1, A_2, \ldots, A_n$ be attributes with domains $D_1, D_2, \ldots, D_n$. Relation R is a set of n-tuples:
      - – $(A_1{:}d_1, A_2{:}d_2, \ldots, A_n{:}d_n)$ such that $d_1 \in D_1, d_2 \in D_2, \ldots, d_n \in D_n$
- Relational database schema
  - – A set of relation schemas, each with a distinct name.



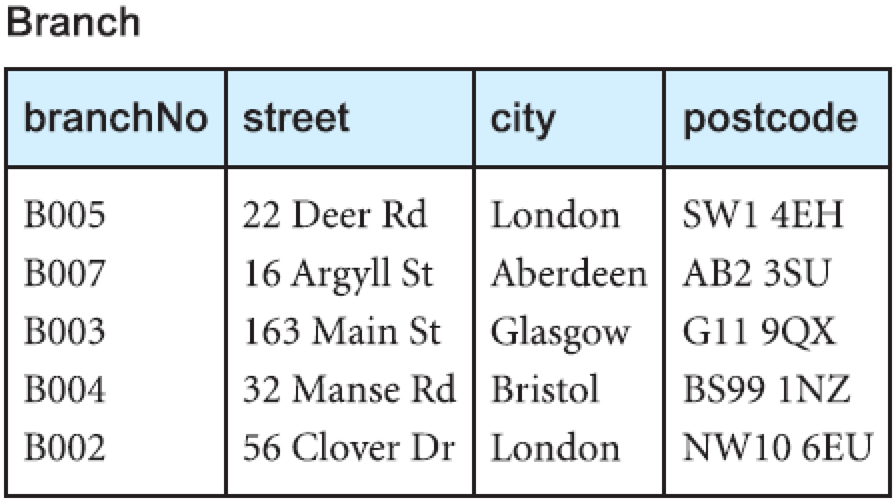
Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

{
(branchNo: B005, street: 22 Deer Rd, city: London, postcode: SW1 4EH),
(branchNo: B007, street: 16 Argyll St, city: Aberdeen, postcode: AB2 3SU),
(branchNo: B003, street: 163 Main St, city: Glasgow, postcode: G11 9QX),
(branchNo: B004, street: 32 Manse Rd, city: Bristol, postcode: BS99 INZ),
(branchNo: B002, street: 56 Clover Dr, city: London, postcode: NW10 6EU)
}

Relational Keys

- Superkey 由它可以唯一地确定一个对象
  - An attribute, or set of attributes, that uniquely identifies a tuple within a relation.
- Candidate key
  - A superkey such that no proper subset is a superkey within the relation. 在这个表里，除了city以外的任何一个

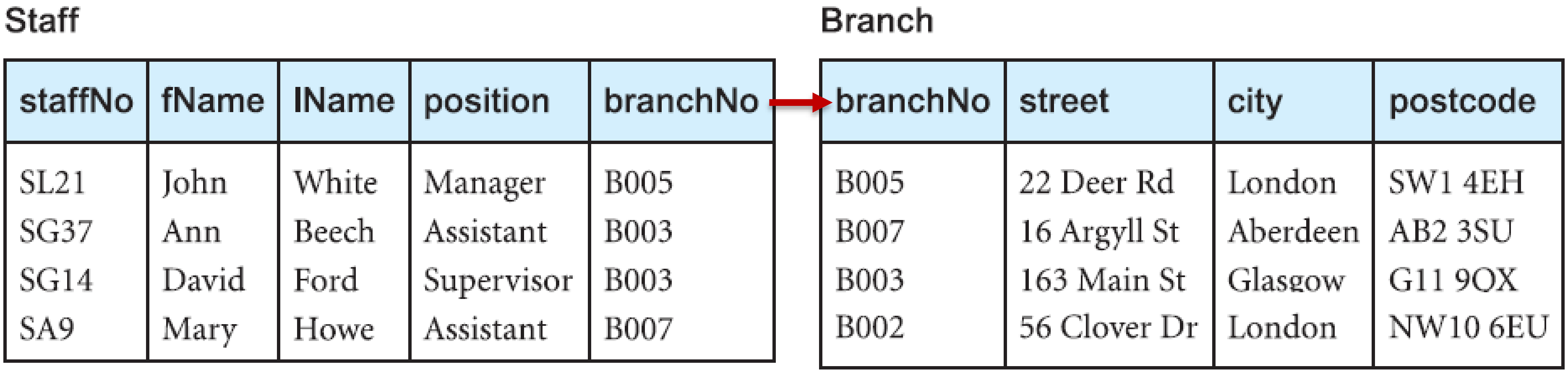


Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

Relational Keys (Cont.)

- Primary key 主键
  - The candidate key that is selected to identify tuples uniquely within the key relation
- Foreign key 外键：这个键还会放在其他的关系中，来建立联系 如果有值，常用来建立constrains约束
  - An attribute, or set of attributes, within one relation that matches the primary key of some (possibly the same) relation.



Staff

staffNo	fName	lName	position	branchNo
SL21	John	White	Manager	B005
SG37	Ann	Beech	Assistant	B003
SG14	David	Ford	Supervisor	B003
SA9	Mary	Howe	Assistant	B007

Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B002	56 Clover Dr	London	NW10 6EU

View

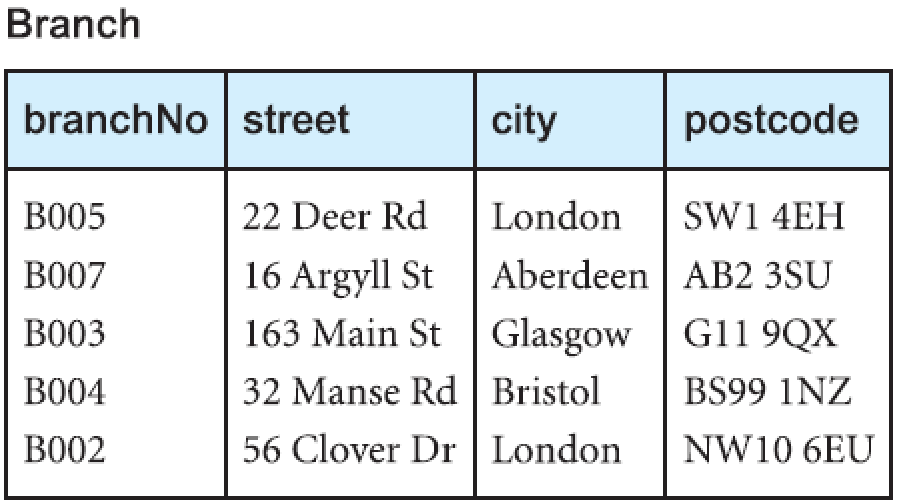
- Base relation
  - A named relation corresponding to an entity in the conceptual schema, whose tuples are physically stored in the database
- View

  这个view基本上是，在basic relation上计算得到的结果
  需要的时候才计算，并不总是会存在数据库里
  - The dynamic result of one or more relational operations operating on the base relations to produce another relation.
  - A *virtual relation* that does not necessarily exist in the database but can be produced upon request by a particular user, at the time of request.

Integrity Constraints

- Null 约束可以是空集
  - Represents a value for an attribute that is currently unknown or is not applicable for this tuple.
- Entity integrity
  - In a base relation, no attribute of a primary key can be null
- Referential integrity
  - If a foreign key exists in a relation, either the foreign key value must match a primary key value of some tuple in its home relation or the foreign key value must be wholly null.
- General constraints
  - Additional rules specified by the users or database administrators of a database that define or constrain some aspect of the enterprise.

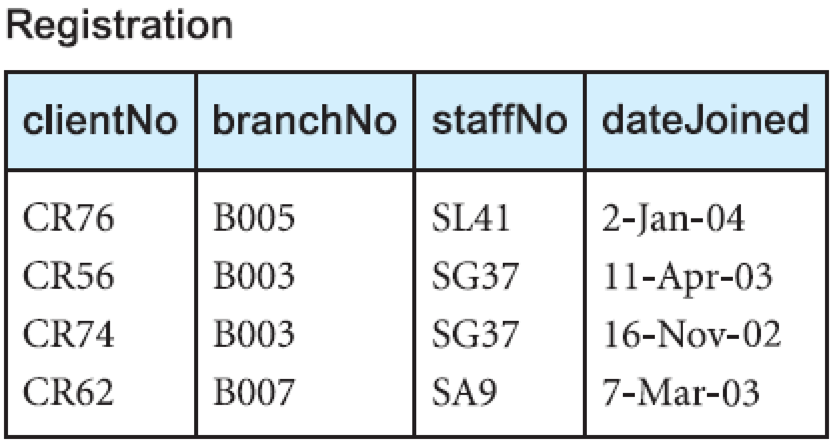
Representing Relational Database Schemas



Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

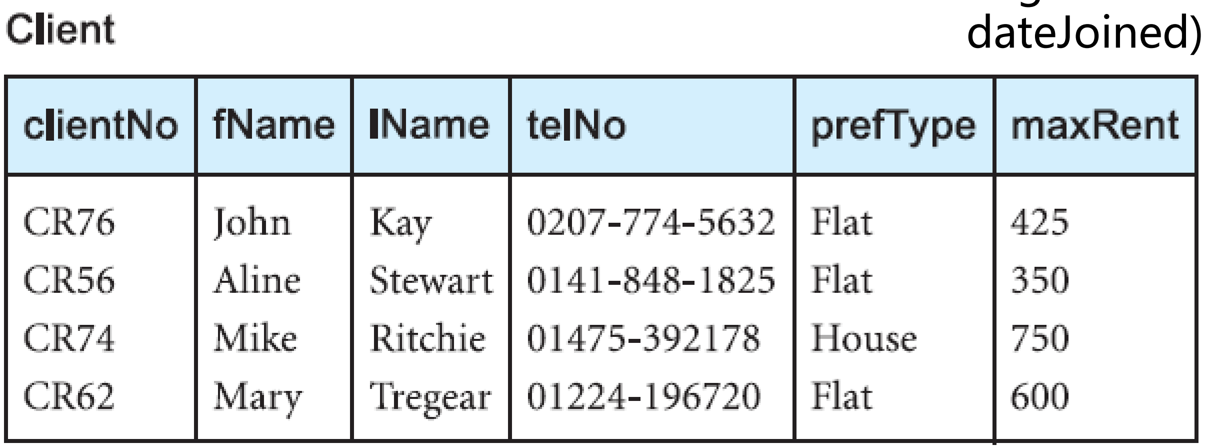
Branch (branchNo, street, city, postcode)



Registration

clientNo	branchNo	staffNo	dateJoined
CR76	B005	SL41	2-Jan-04
CR56	B003	SG37	11-Apr-03
CR74	B003	SG37	16-Nov-02
CR62	B007	SA9	7-Mar-03

Registration (clientNo, branchNo, staffNo, dateJoined)



Client

clientNo	fName	lName	telNo	prefType	maxRent
CR76	John	Kay	0207-774-5632	Flat	425
CR56	Aline	Stewart	0141-848-1825	Flat	350
CR74	Mike	Ritchie	01475-392178	House	750
CR62	Mary	Tregear	01224-196720	Flat	600

Client (clientNo, fName, lName, telNo, prefType, maxRent)

Relational Algebra

- Theoretical way of manipulating table contents using relational operators
  - Relvar: variable that holds a relation
    - Heading contains the names of the attributes
    - Body contains the relation
  - Relational operators have the property of closure
    - Closure: use of relational algebra operators on existing relations produces new relations

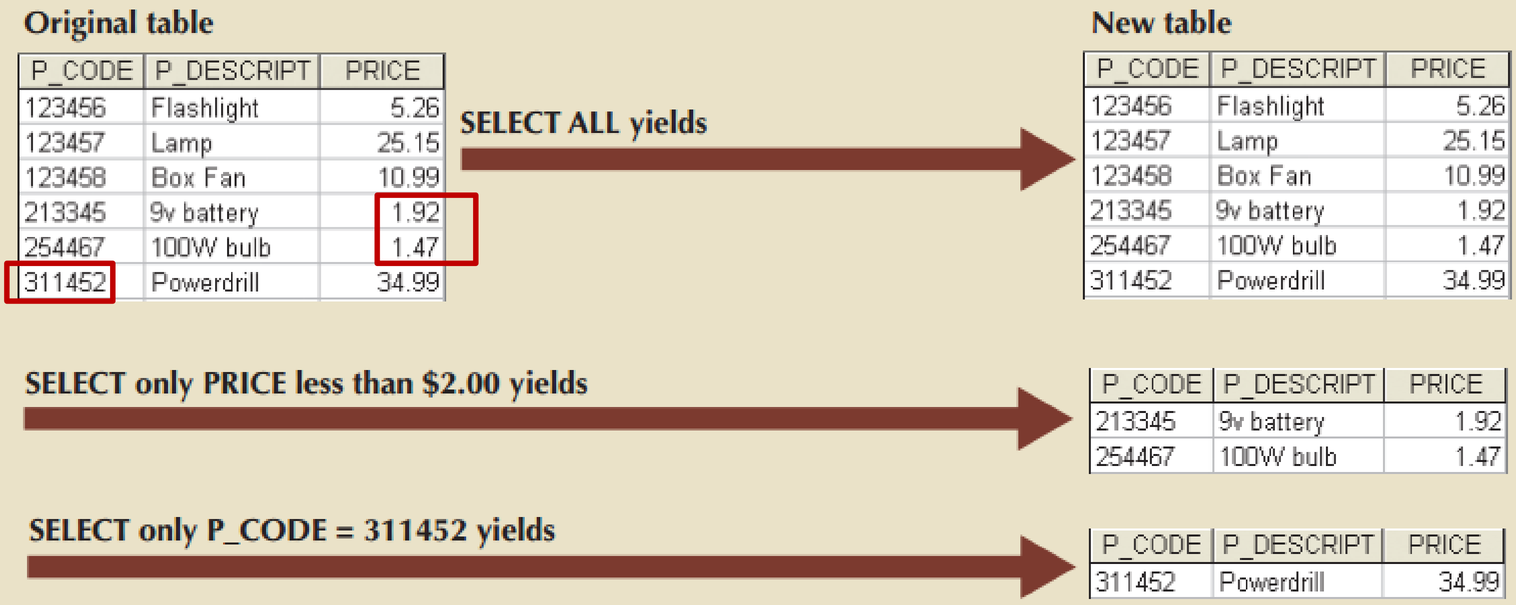
封闭性

Relational Set Operators (1 of 13)

- Select (restrict)
  - Unary operator that yields a horizontal subset of a table
- Project
  - Unary operator that yields a vertical subset of a table
- Union
  - Combines all rows from two tables, excluding duplicate rows
  - Union-compatible: tables share the same number of columns, and their corresponding columns share compatible domains
- Intersect
  - Yields only the rows that appear in both tables
  - Tables must be union-compatible to yield valid results

Relational Set Operators (2 of 13)

FIGURE 3.4 SELECT



Original table

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

SELECT ALL yields

New table

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

SELECT only PRICE less than $2.00 yields

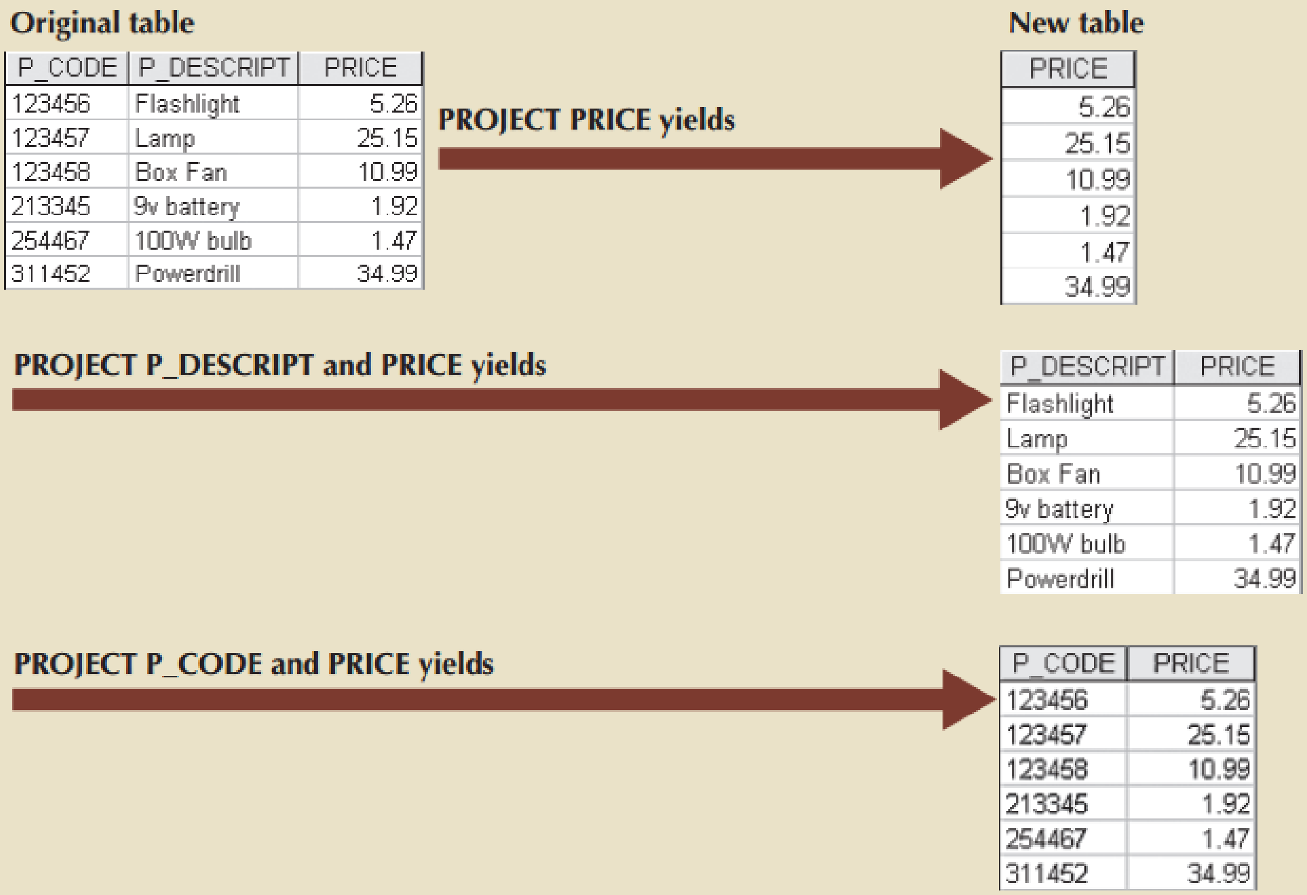
P_CODE	P_DESCRIPT	PRICE
213345	9v battery	1.92
254467	100W bulb	1.47

SELECT only P_CODE = 311452 yields

P_CODE	P_DESCRIPT	PRICE
311452	Powerdrill	34.99

Relational Set Operators (3 of 13)

FIGURE 3.5 PROJECT



Original table

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

PROJECT PRICE yields

New table

PRICE
5.26
25.15
10.99
1.92
1.47
34.99

PROJECT P_DESCRIPT and PRICE yields

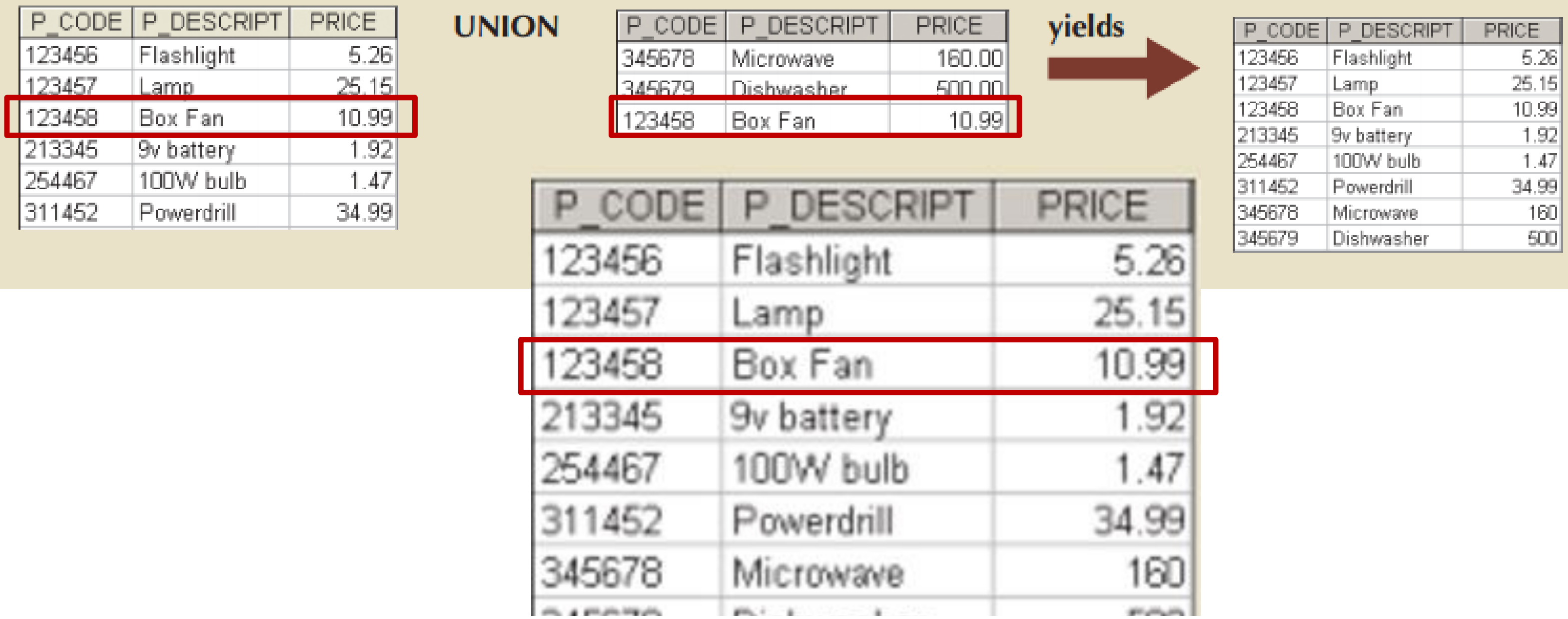
P_DESCRIPT	PRICE
Flashlight	5.26
Lamp	25.15
Box Fan	10.99
9v battery	1.92
100W bulb	1.47
Powerdrill	34.99

PROJECT P_CODE and PRICE yields

P_CODE	PRICE
123456	5.26
123457	25.15
123458	10.99
213345	1.92
254467	1.47
311452	34.99

Relational Set Operators (4 of 13)

FIGURE 3.6 UNION



P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

UNION

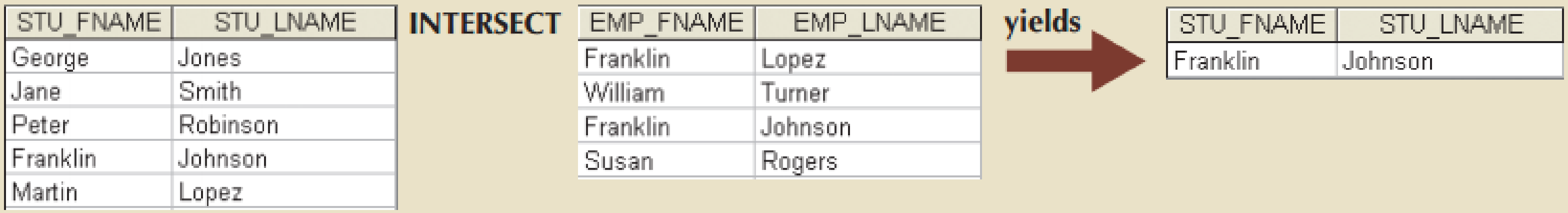
P_CODE	P_DESCRIPT	PRICE
345678	Microwave	160.00
345679	Dishwasher	500.00
123458	Box Fan	10.99

yields

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99
345678	Microwave	160
345679	Dishwasher	500

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99
345678	Microwave	160

FIGURE 3.7 INTERSECT



STU_FNAME	STU_LNAME
George	Jones
Jane	Smith
Peter	Robinson
Franklin	Johnson
Martin	Lopez

INTERSECT

EMP_FNAME	EMP_LNAME
Franklin	Lopez
William	Turner
Franklin	Johnson
Susan	Rogers

yields

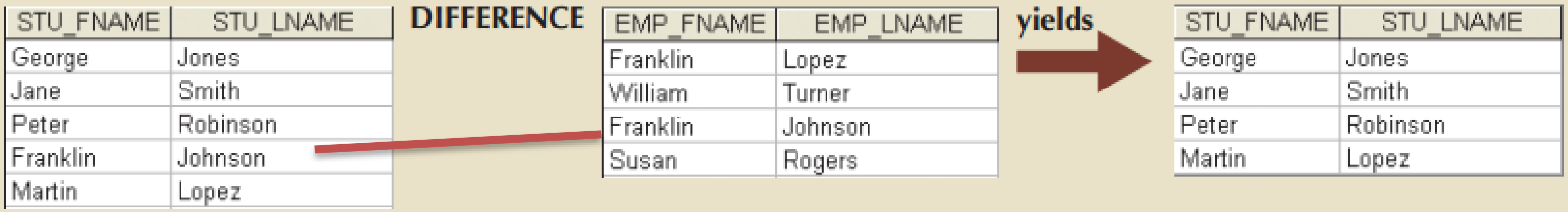
STU_FNAME	STU_LNAME
Franklin	Johnson

Relational Set Operators (5 of 13)

- Difference
  - Yields all rows in one table that are not found in the other table
  - Tables must be union-compatible to yield valid results
- Product
  - Yields all possible pairs of rows from two tables

Relational Set Operators (6 of 13)

FIGURE 3.8 DIFFERENCE



STU_FNAME	STU_LNAME
George	Jones
Jane	Smith
Peter	Robinson
Franklin	Johnson
Martin	Lopez

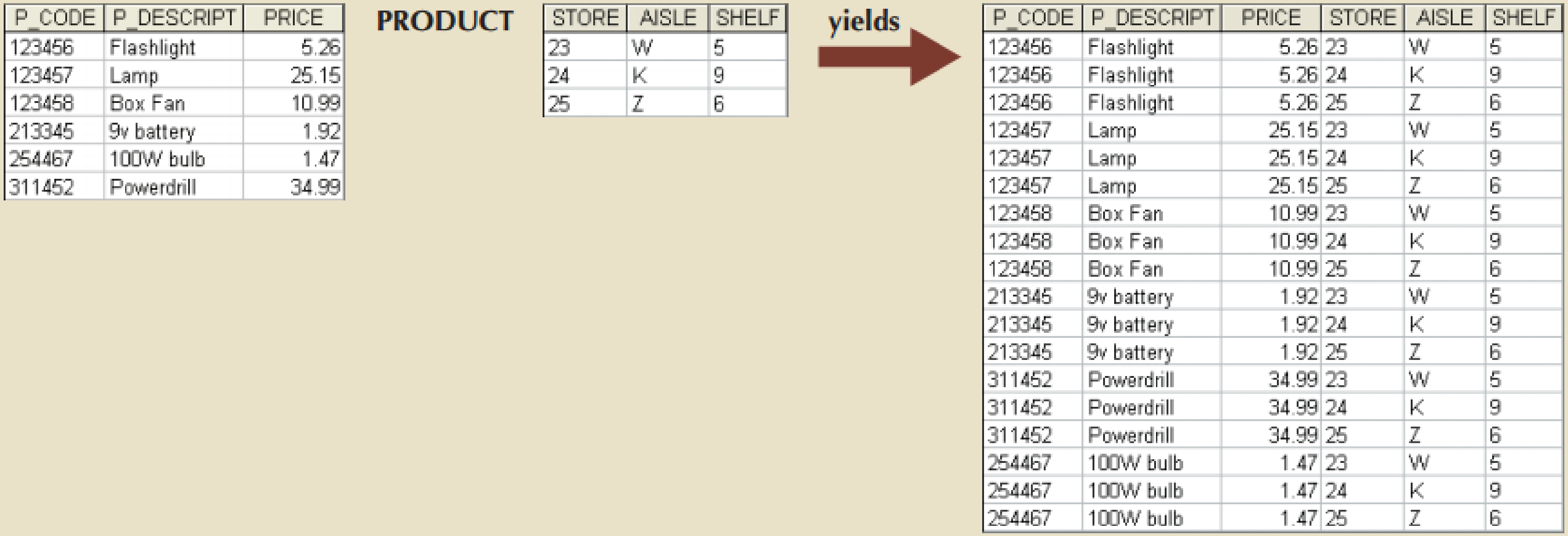
DIFFERENCE

EMP_FNAME	EMP_LNAME
Franklin	Lopez
William	Turner
Franklin	Johnson
Susan	Rogers

yields

STU_FNAME	STU_LNAME
George	Jones
Jane	Smith
Peter	Robinson
Martin	Lopez

FIGURE 3.9 PRODUCT



P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

PRODUCT

STORE	AISLE	SHELF
23	W	5
24	K	9
25	Z	6

yields

P_CODE	P_DESCRIPT	PRICE	STORE	AISLE	SHELF
123456	Flashlight	5.26	23	W	5
123456	Flashlight	5.26	24	K	9
123456	Flashlight	5.26	25	Z	6
123457	Lamp	25.15	23	W	5
123457	Lamp	25.15	24	K	9
123457	Lamp	25.15	25	Z	6
123458	Box Fan	10.99	23	W	5
123458	Box Fan	10.99	24	K	9
123458	Box Fan	10.99	25	Z	6
213345	9v battery	1.92	23	W	5
213345	9v battery	1.92	24	K	9
213345	9v battery	1.92	25	Z	6
311452	Powerdrill	34.99	23	W	5
311452	Powerdrill	34.99	24	K	9
311452	Powerdrill	34.99	25	Z	6
254467	100W bulb	1.47	23	W	5
254467	100W bulb	1.47	24	K	9
254467	100W bulb	1.47	25	Z	6

Relational Set Operators (7 of 13)

- Joins allow information to be intelligently combined from two or more tables
  - Natural join: links tables by selecting only the rows with common values in their common attribute
  - Equijoin: links tables on the basis of an equality condition that compares specified columns of each table
  - Theta join: links tables using an inequality comparison operator
  - Inner join: only returns matched records from the tables that are being joined
  - Outer join: matched pairs are retained and unmatched values in the other table are left null
    - Left outer join: yields all of the rows in the first table, including those that do not have a matching value in the second table
    - Right outer join: yields all of the rows in the second table, including those that do not have matching values in the first table

Relational Set Operators (8 of 13)

FIGURE 3.10 TWO TABLES THAT WILL BE USED IN JOIN ILLUSTRATIONS

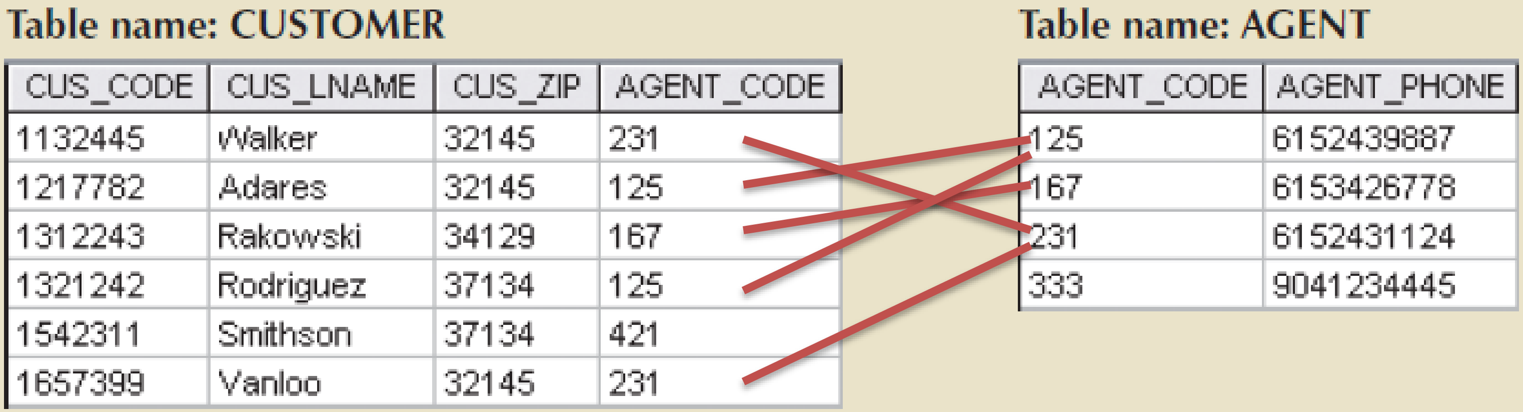
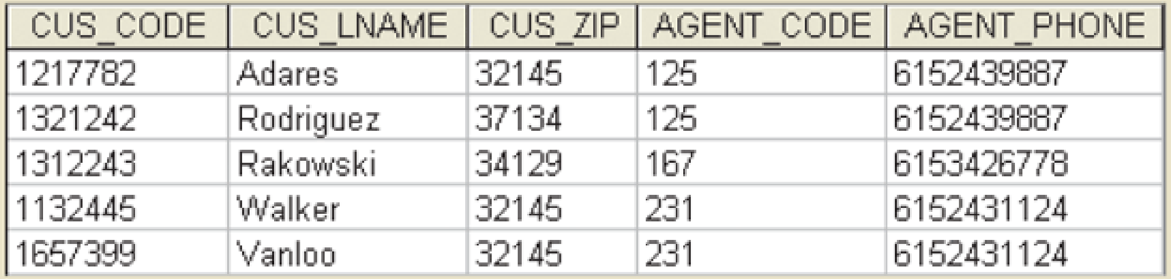


Table name: CUSTOMER

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE
1132445	Walker	32145	231
1217782	Adares	32145	125
1312243	Rakowski	34129	167
1321242	Rodriguez	37134	125
1542311	Smithson	37134	421
1657399	Vanloo	32145	231

Table name: AGENT

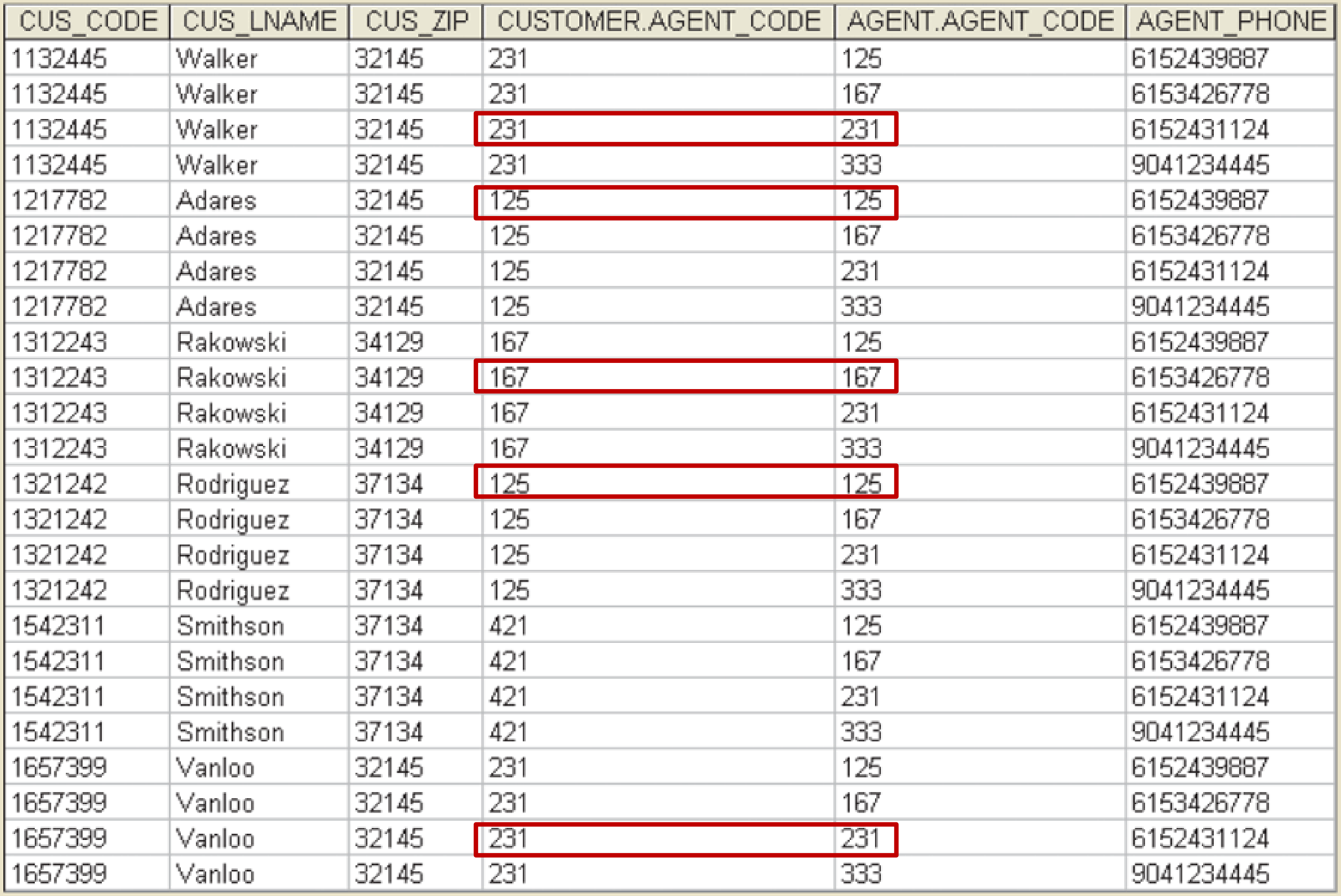
AGENT_CODE	AGENT_PHONE
125	6152439887
167	6153426778
231	6152431124
333	9041234445



CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	6152439887
1321242	Rodriguez	37134	125	6152439887
1312243	Rakowski	34129	167	6153426778
1132445	Walker	32145	231	6152431124
1657399	Vanloo	32145	231	6152431124

Relational Set Operators (9 of 13)

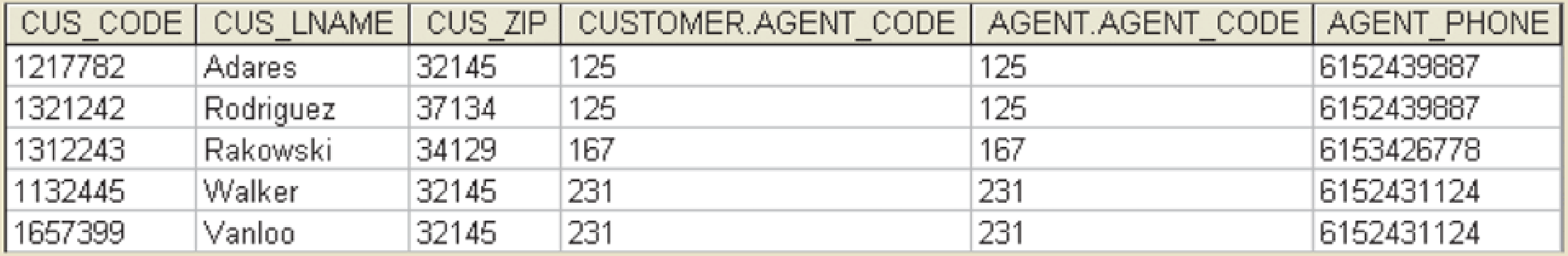
FIGURE 3.11 NATURAL JOIN, STEP 1: PRODUCT



CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1132445	Walker	32145	231	125	6152439887
1132445	Walker	32145	231	167	6153426778
1132445	Walker	32145	231	231	6152431124
1132445	Walker	32145	231	333	9041234445
1217782	Adares	32145	125	125	6152439887
1217782	Adares	32145	125	167	6153426778
1217782	Adares	32145	125	231	6152431124
1217782	Adares	32145	125	333	9041234445
1312243	Rakowski	34129	167	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1312243	Rakowski	34129	167	231	6152431124
1312243	Rakowski	34129	167	333	9041234445
1321242	Rodriguez	37134	125	125	6152439887
1321242	Rodriguez	37134	125	167	6153426778
1321242	Rodriguez	37134	125	231	6152431124
1321242	Rodriguez	37134	125	333	9041234445
1542311	Smithson	37134	421	125	6152439887
1542311	Smithson	37134	421	167	6153426778
1542311	Smithson	37134	421	231	6152431124
1542311	Smithson	37134	421	333	9041234445
1657399	Vanloo	32145	231	125	6152439887
1657399	Vanloo	32145	231	167	6153426778
1657399	Vanloo	32145	231	231	6152431124
1657399	Vanloo	32145	231	333	9041234445

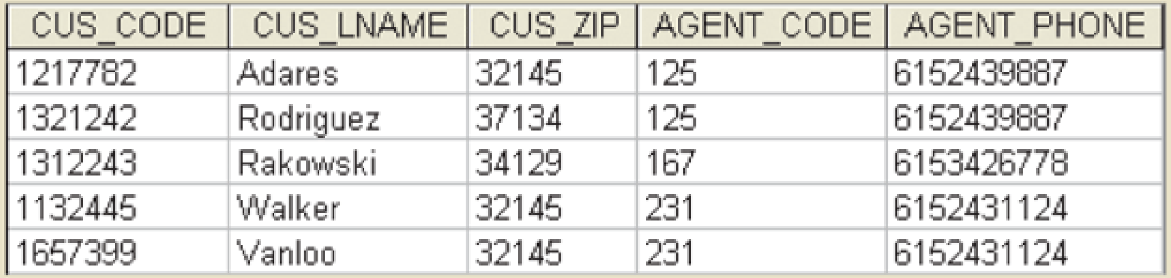
Relational Set Operators (10 of 13)

FIGURE 3.12 NATURAL JOIN, STEP 2: SELECT



CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	125	6152439887
1321242	Rodriguez	37134	125	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1132445	Walker	32145	231	231	6152431124
1657399	Vanloo	32145	231	231	6152431124

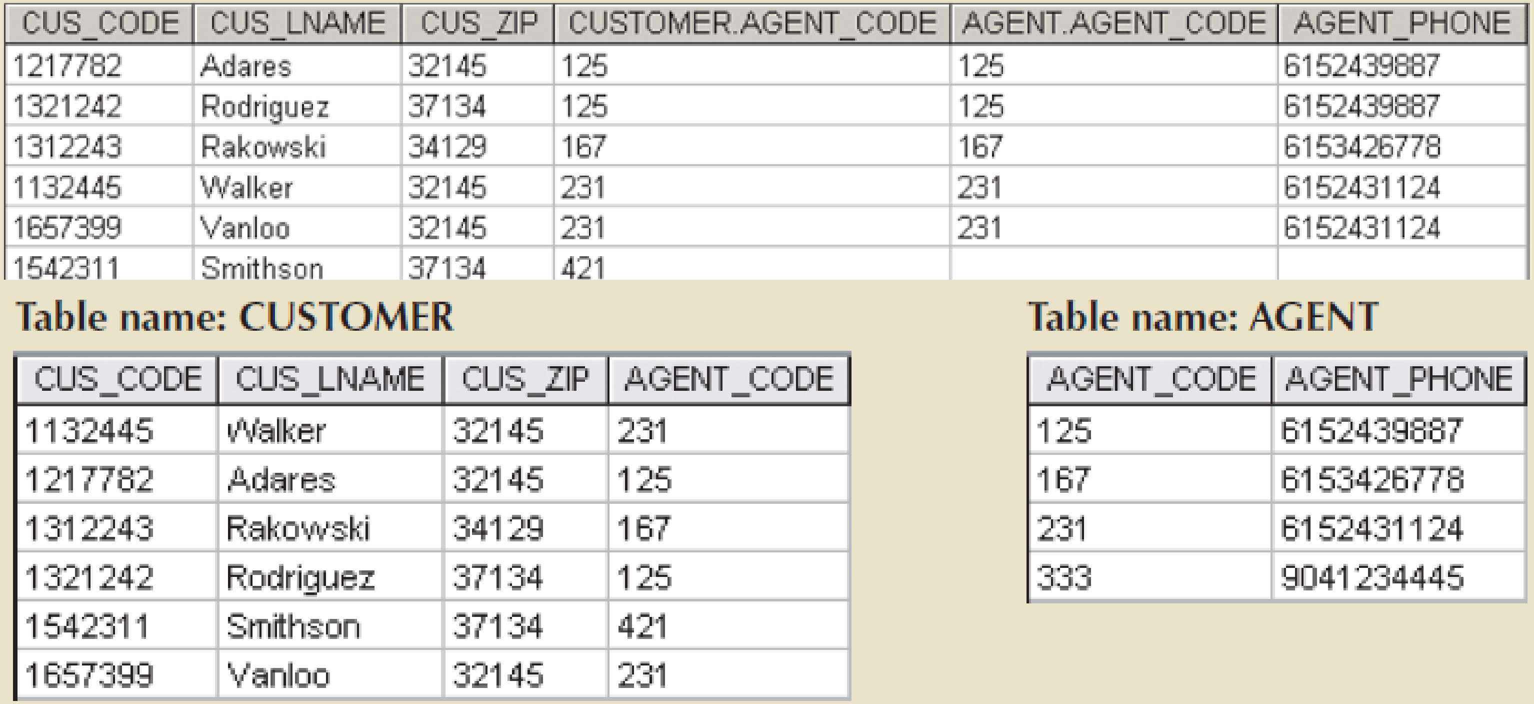
FIGURE 3.13 NATURAL JOIN, STEP 3: PROJECT



CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	6152439887
1321242	Rodriguez	37134	125	6152439887
1312243	Rakowski	34129	167	6153426778
1132445	Walker	32145	231	6152431124
1657399	Vanloo	32145	231	6152431124

Relational Set Operators (11 of 13)

FIGURE 3.14 LEFT OUTER JOIN



CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	125	6152439887
1321242	Rodriguez	37134	125	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1132445	Walker	32145	231	231	6152431124
1657399	Vanloo	32145	231	231	6152431124
1542311	Smithson	37134	421		

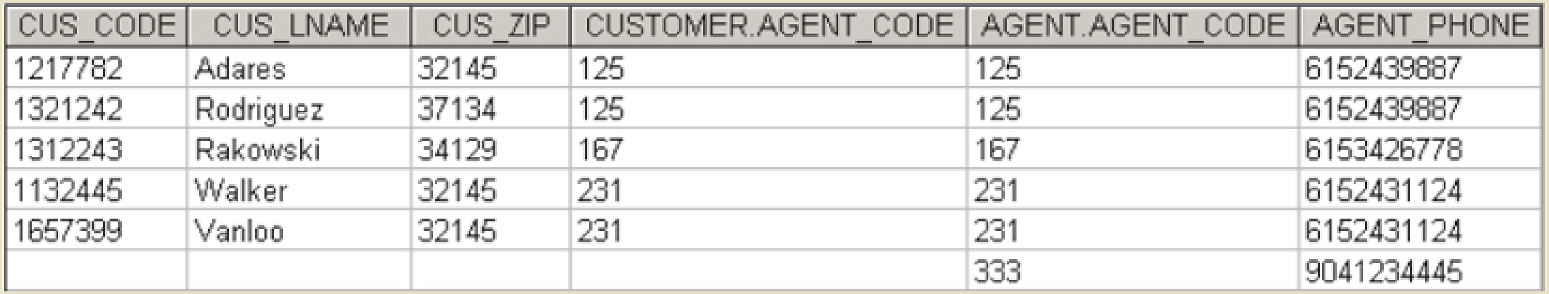
Table name: CUSTOMER

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE
1132445	Walker	32145	231
1217782	Adares	32145	125
1312243	Rakowski	34129	167
1321242	Rodriguez	37134	125
1542311	Smithson	37134	421
1657399	Vanloo	32145	231

Table name: AGENT

AGENT_CODE	AGENT_PHONE
125	6152439887
167	6153426778
231	6152431124
333	9041234445

FIGURE 3.15 RIGHT OUTER JOIN



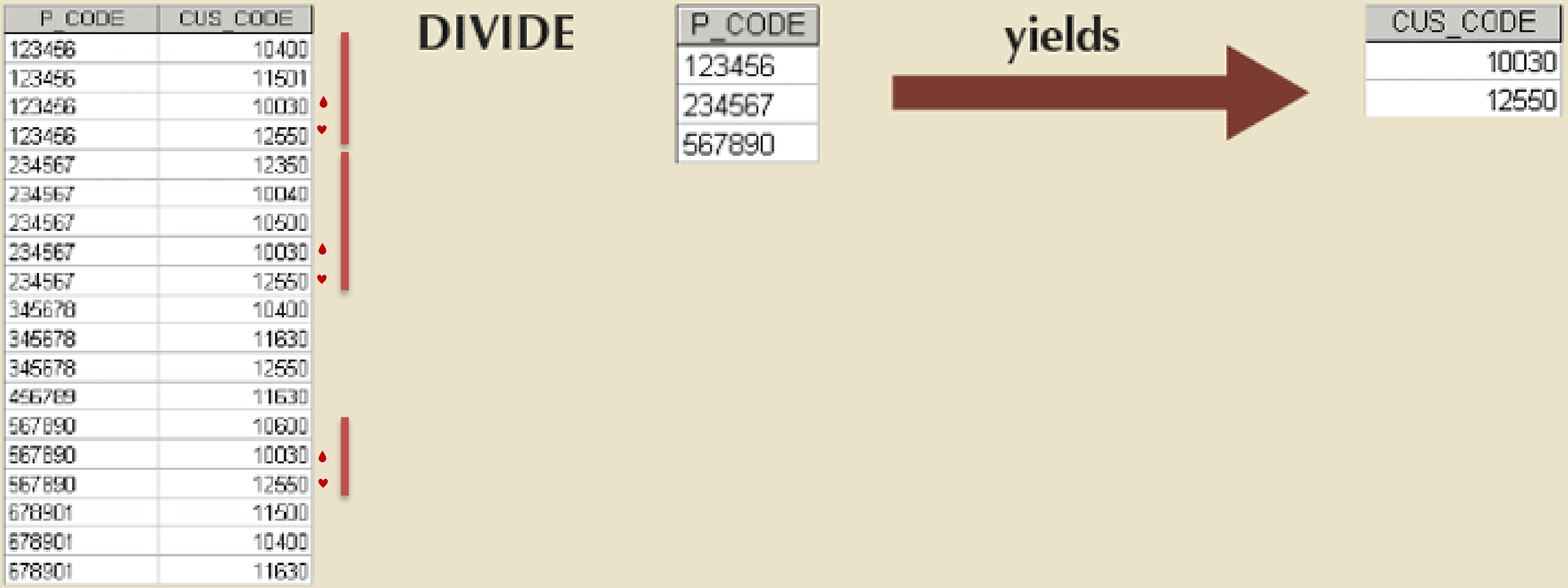
CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	125	6152439887
1321242	Rodriguez	37134	125	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1132445	Walker	32145	231	231	6152431124
1657399	Vanloo	32145	231	231	6152431124
				333	9041234445

Relational Set Operators (12 of 13)

- Divide
  - Uses one double-column table as the dividend and one single-column table as the divisor
  - Output is a single column that contains all values from the second column of the dividend that are associated with every row in the divisor

Relational Set Operators (13 of 13)

FIGURE 3.16 DIVIDE



P_CODE	CUS_CODE
123456	10400
123456	11501
123456	10030
123456	12550
234567	12350
234567	10040
234567	10500
234567	10030
234567	12550
345678	10400
345678	11630
345678	12550
456789	11630
567890	10600
567890	10030
567890	12550
678901	11500
678901	10400
678901	11630

DIVIDE

P_CODE
123456
234567
567890

yields

CUS_CODE
10030
12550

Data Dictionary and the System Catalog

- Data dictionary
  - – Description of all tables in the database created by the user and designer
- System catalog
  - – System data dictionary that describes all objects within the database
- Homonyms and synonyms must be avoided to lessen confusion
  - Homonym: same name is used to label different attributes
  - Synonym: different names are used to describe the same attribute

RDBMS: Pros & Cons

Pros

- Strong theoretical foundation
- Declarative syntax
- Standard data access language through SQL
- …

Cons

- Scalability
- Data has to fit into tables
- …

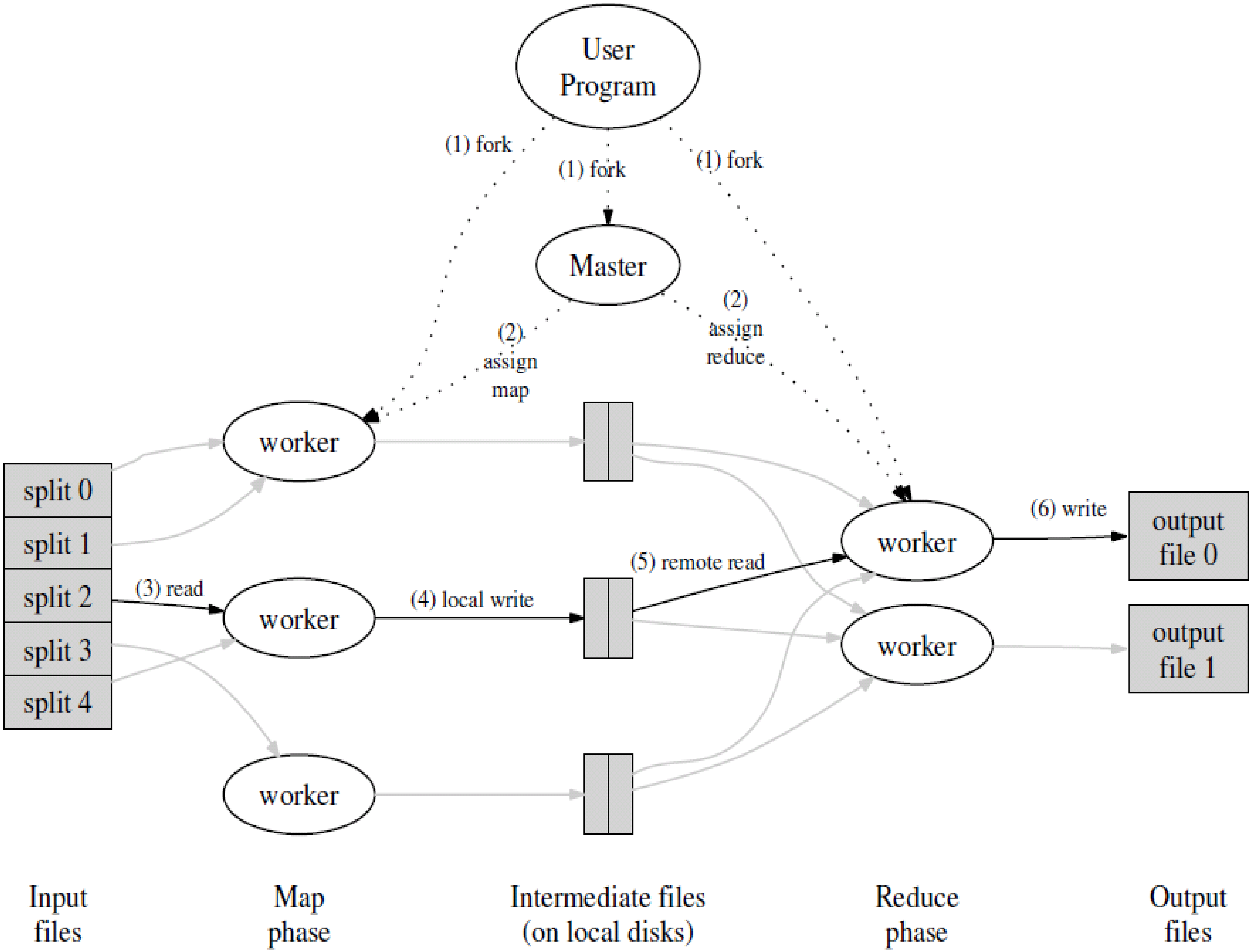
Big Data

- Other characteristics beyond 3Vs
  - Value: degree data can be analyzed for meaningful insight
  - Veracity: trustworthiness of data
  - Variability: changes in meaning of data based on context
  - Sentimental analysis: attempts to determine if a statement conveys a positive, negative, or neutral attitude about a topic
  - Visualization: ability to graphically resent data to make it understandable

Hadoop

- De facto standard for most Big Data storage and processing
  - Java-based framework for distributing and processing very large data sets across clusters of computers
- Important components
  - Hadoop Distributed File System (HDFS): low-level distributed file processing system that can be used directly for data storage
  - MapReduce: programming model that supports processing large data sets

Execution procedure



Case study

- We have a large file of words, one word to a line
- Count the number of times each distinct word appears in the file

- *Sample application*: analyze web server logs to find popular URLs

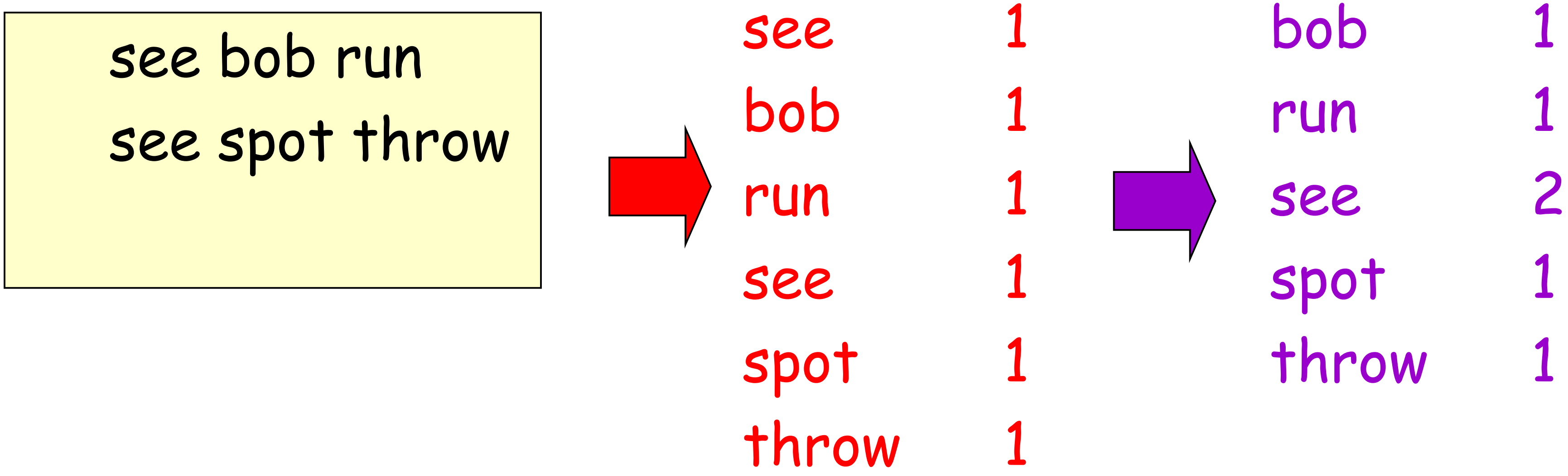
Word Count using MapReduce

```
map(key, value):
// key: document name; value: text of document
    for each word w in value:
        emit(w, 1)
```

```
reduce(key, values):
// key: a word; values: an iterator over counts
        result = 0
        for each count v in values:
            result += v
        emit(key, result)
```

Word Count, illustrated

```
map(key=url, val=contents):
  For each word w in contents, emit (w, “1”)
reduce(key=word, values=uniq_counts):
      Sum all “1”s in values list
      Emit result “(word, sum)”
```



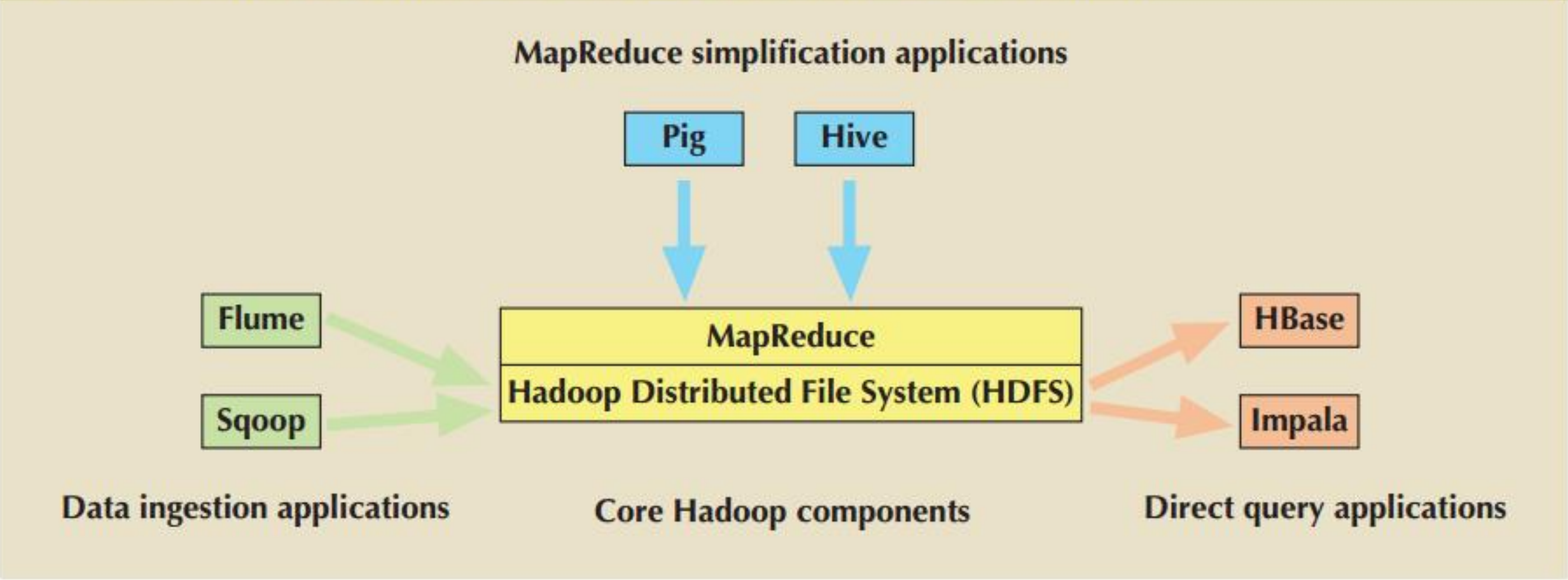
see bob run
see spot throw

see	1
bob	1
run	1
see	1
spot	1
throw	1

bob	1
run	1
see	2
spot	1
throw	1

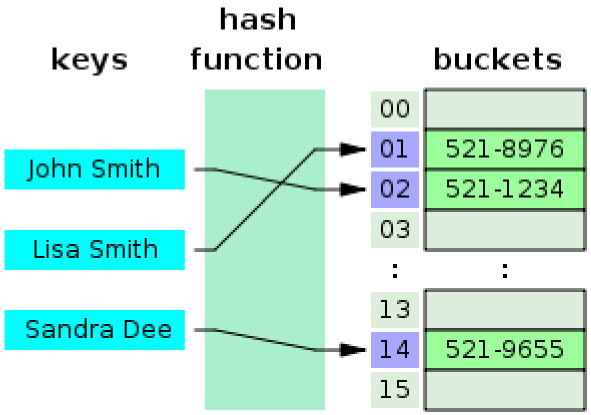
Hadoop Ecosystem

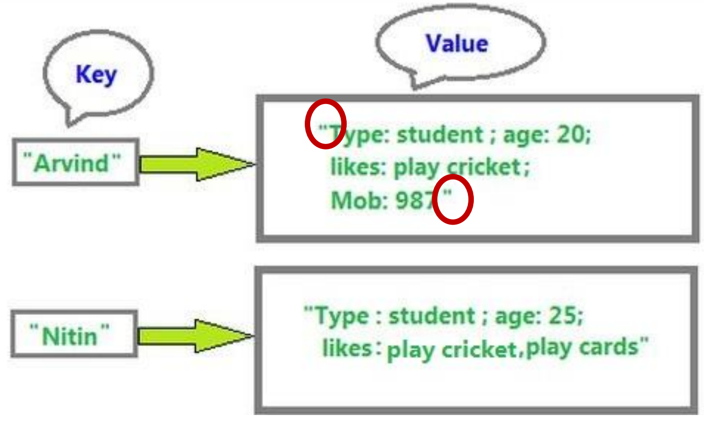
FIGURE 14.6 A SAMPLE OF THE HADOOP ECOSYSTEM

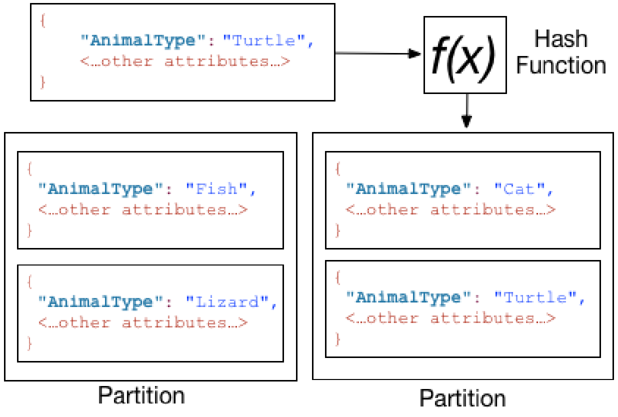

MapReduce simplification applications
Pig
Hive
Flume
Sqoop
MapReduce
Hadoop Distributed File System (HDFS)
HBase
Impala
Data ingestion applications
Core Hadoop components
Direct query applications

No SQL—Not Only SQL

- Key-value DBMS
  - Schema free
  - High Scalability




Key
Value
"Arvind"
"Type: student ; age: 20;
likes: play cricket;
Mob: 987"
"Nitin"
"Type : student ; age: 25;
likes: play cricket,play cards"

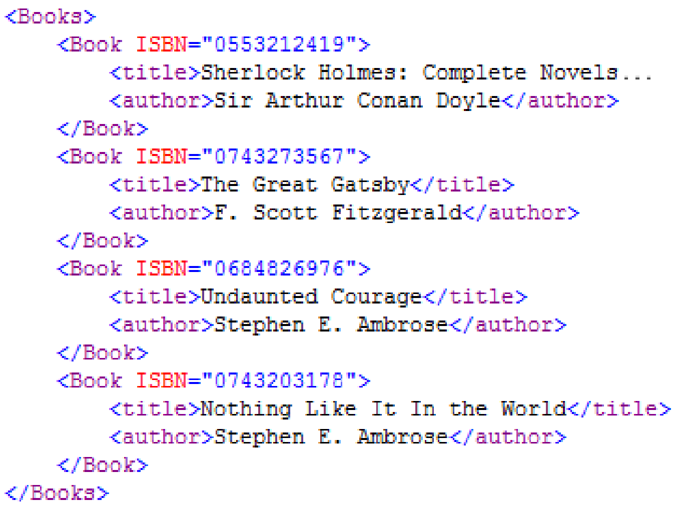


redis

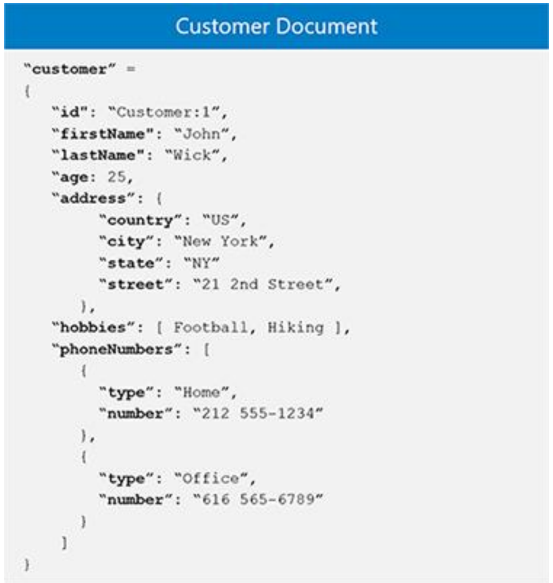
amazon DynamoDB

No SQL—Not Only SQL

- Document-oriented DBMS
  - Self-describing
  - Hierarchical tree data structure
  - Documents have differences in their attributes
    - But belong to same collection



```
<Books>
    <Book ISBN="0553212419">
        <title>Sherlock Holmes: Complete Novels...
        <author>Sir Arthur Conan Doyle</author>
    </Book>
    <Book ISBN="0743273567">
        <title>The Great Gatsby</title>
        <author>F. Scott Fitzgerald</author>
    </Book>
    <Book ISBN="0684826976">
        <title>Undaunted Courage</title>
        <author>Stephen E. Ambrose</author>
    </Book>
    <Book ISBN="0743203178">
        <title>Nothing Like It In the World</title>
        <author>Stephen E. Ambrose</author>
    </Book>
</Books>
```



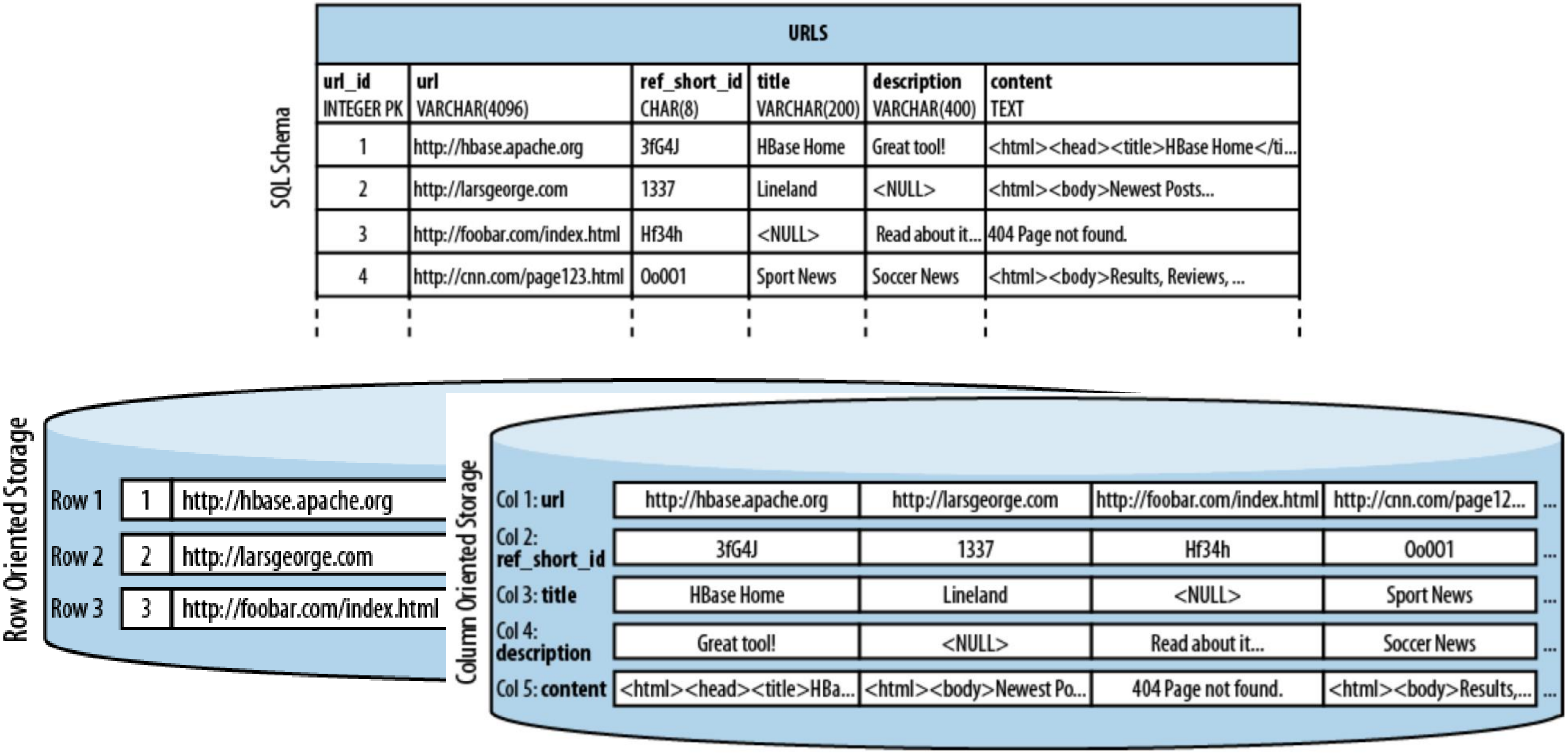
Customer Document

```
"customer" =
{
    "id": "Customer:1",
    "firstName": "John",
    "lastName": "Wick",
    "age: 25,
    "address": {
         "country": "US",
         "city": "New York",
         "state": "NY"
         "street": "21 2nd Street",
      },
    "hobbies": [ Football, Hiking ],
    "phoneNumbers": [
      {
         "type": "Home",
         "number": "212 555-1234"
      },
      {
         "type": "Office",
         "number": "616 565-6789"
      }
     ]
}
```

mongoDB

No SQL—Not Only SQL

- Column-oriented DBMS



SQL Schema

URLS					
url_id INTEGER PK	url VARCHAR(4096)	ref_short_id CHAR(8)	title VARCHAR(200)	description VARCHAR(400)	content TEXT
1	http://hbase.apache.org	3fG4J	HBase Home	Great tool!	<html><head><title>HBase Home</ti...
2	http://larsgeorge.com	1337	Lineland	<NULL>	<html><body>Newest Posts...
3	http://foobar.com/index.html	Hf34h	<NULL>	Read about it...	404 Page not found.
4	http://cnn.com/page123.html	0o001	Sport News	Soccer News	<html><body>Results, Reviews, ...

Row Oriented Storage

Row 1	1	http://hbase.apache.org
Row 2	2	http://larsgeorge.com
Row 3	3	http://foobar.com/index.html

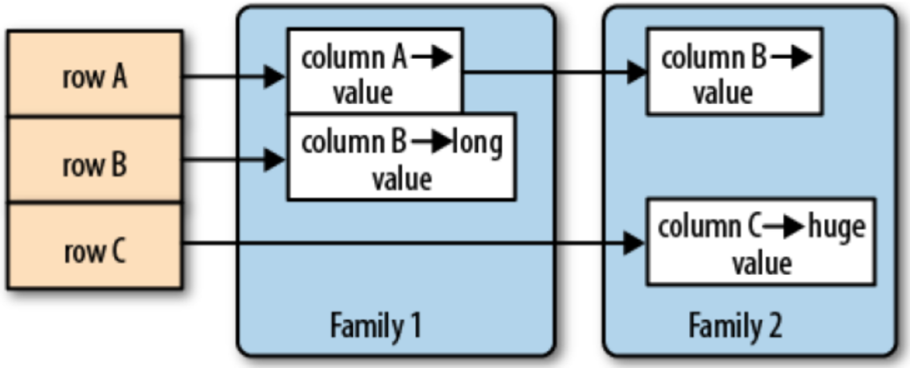
Column Oriented Storage

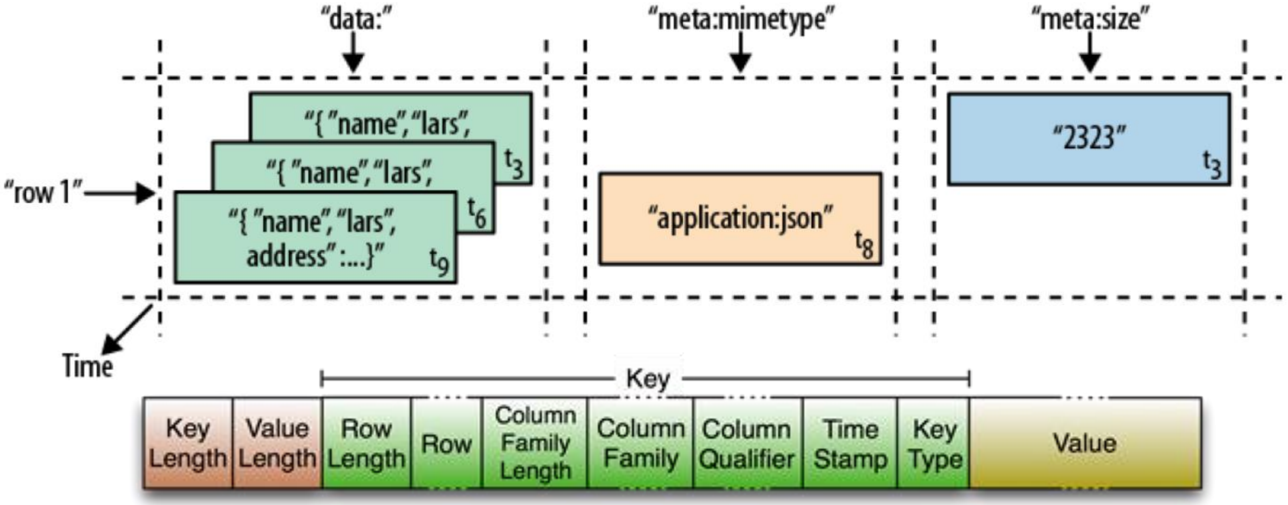
Col 1: **url**	http://hbase.apache.org	http://larsgeorge.com	http://foobar.com/index.html	http://cnn.com/page12...	...
Col 2: **ref_short_id**	3fG4J	1337	Hf34h	0o001	...
Col 3: **title**	HBase Home	Lineland	<NULL>	Sport News	...
Col 4: **description**	Great tool!	<NULL>	Read about it...	Soccer News	...
Col 5: **content**	<html><head><title>HBa...	<html><body>Newest Po...	404 Page not found.	<html><body>Results,...	...

No SQL—Not Only SQL

- Column-oriented DBMS

	column A (int)	column B (varchar)	column C (boolean)	column D (date)
row A				
row B				
row C			NULL?	
row D				





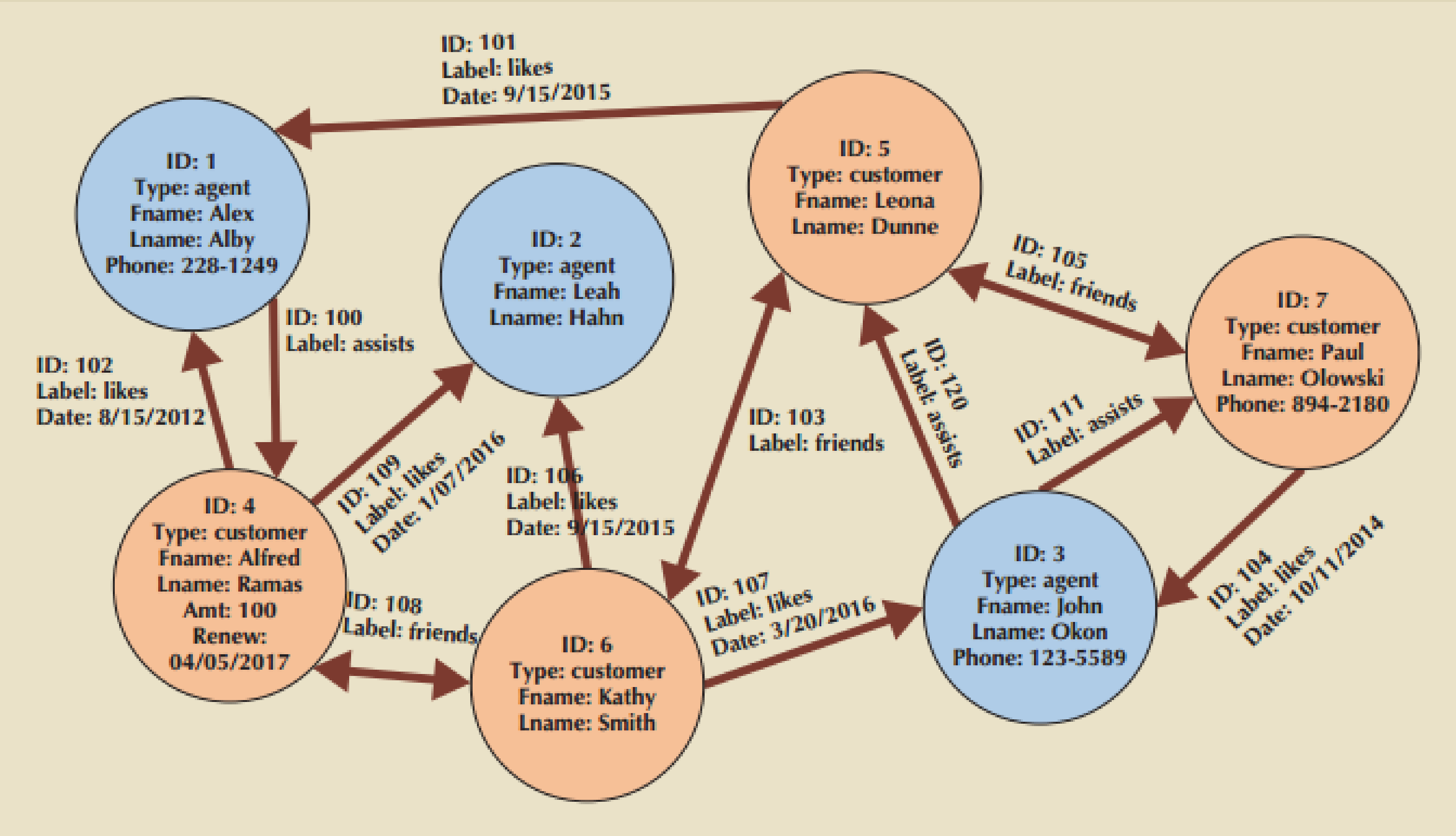
cassandra

APACHE HBASE

No SQL—Not Only SQL

- Graph DBMS

FIGURE 14.11 GRAPH DATABASE REPRESENTATION



NewSQL Databases

- NewSQL databases support:
  - SQL as the primary interface
  - ACID-compliant transactions
- Similar to NoSQL, NewSQL databases also support:
  - Highly distributed clusters
  - Key-value or column-oriented data stores
- Latest technologies to address Big Data problems
  - Have been adopted by relatively few organizations