Physical Database Design

University of California, Berkeley

School of Information

INFO 257: Database Management

Announcements

- Welcome Back!
  - Questions/Comments?
- Assignment 3 Due Next Week (See github for specifics)
- Assignment 4 (Final Project) Released (also in github)
- Lecture: Physical DB Design
- Workshop: Assignment 4 Review, Logical Backups

Integrated Data Management Framework

<table>
<tr><th></th><th>Operational</th><th colspan="2">Informational</th></tr>
<tr><td></td><td>Transactional</td><td>Analytical–
Data Warehousing</td><td>Analytical–
Big Data</td></tr>
<tr><td>Technology</td><td>Relational</td><td>Relational</td><td>Non-relational</td></tr>
<tr><td>Modeling</td><td>Conceptual data modeling with (E)ER (Chapters 2 and 3)</td><td rowspan="3">Data warehousing and data integration (Chapter 9)</td><td rowspan="3">Big data technologies, including Hadoop & NoSQL (Chapter 10)</td></tr>
<tr><td>Design</td><td>Logical data modeling with the relational model; Normalization (Chapter 4)</td></tr>
<tr><td>Infrastructure</td><td>Physical design of relational databases; Security; Cloud computing (Chapter 8)</td></tr>
<tr><td>Access</td><td>SQL (Chapters 5 and 6)

Applications with SQL (Chapter 7)</td><td></td><td></td></tr>
<tr><td>Data analysis</td><td colspan="3">Analytics and its implications (Chapter 11)</td></tr>
<tr><td>Governance and data management</td><td colspan="3">Lifecycle (Chapter 1)
Governance, data quality, and master data management (Chapter 12)</td></tr>
</table>

Lecture Outline

- Physical Database Design
- Access Methods and the Cloud

Lecture Outline

- Physical Database Design
- Access Methods and the Cloud

Physical Database Design

- Purpose – translate the logical description of data into the **technical specifications** for storing and retrieving data
- Goal – create a design for storing data that will provide **adequate performance** and ensure **database integrity**, **security**, and **recoverability**

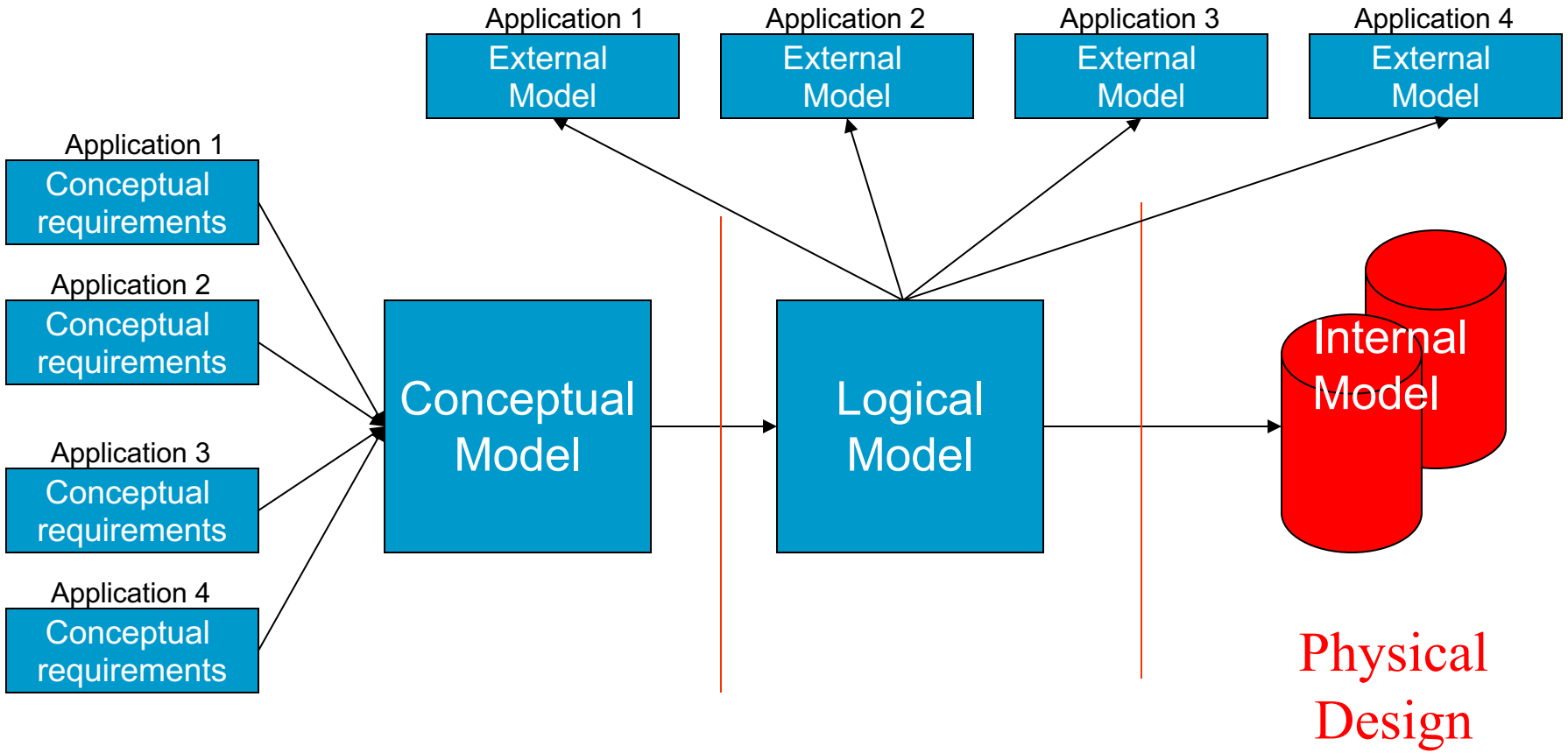
Information Needed for Physical Design

- Normalized relations, including estimates for the range of the number of rows in each table
- Definitions of each attribute, along with physical specifications such as maximum possible length
- Descriptions of where and when data are used in various ways (entered, retrieved, deleted, and updated), including typical frequencies of these events
- Expectations or requirements for response time and data security, backup, recovery, retention, and integrity
- Descriptions of the technologies (database management systems) used for implementing the database

Critical Decisions for Physical Design

- Choosing the storage format (called data type) for each attribute from the logical data model
- Giving the DBMS guidance regarding how to group attributes from the logical data model into physical records
- Giving the DBMS guidance on how to arrange similarly structured records in secondary memory (file organization)
- Selecting structures (including indexes and the overall database architecture) for storing and connecting files to make retrieving related data more efficient
- Preparing strategies for handling queries against the database that will optimize performance (query optimization)

Database Design Process



Physical Database Design

- Many physical database design decisions are implicit in the technology adopted
  - Also, organizations may have standards or an "information architecture" that specifies operating systems, DBMS, and data access languages -- thus constraining the range of possible physical implementations.
- We will be concerned with some of the possible physical implementation issues

Physical Design

- This is where you, as the database designer, are faced with the built-in limitations of whatever DBMS you are using
- You will need to recognize and use, or work around, such limitations

Physical Design for Regulatory Compliance

- Sarbanes- Oxley Act (SOX) – protect investors by improving accuracy and reliability
- Committee of Sponsoring Organizations (COSO) of the Treadway Commission
- IT Infrastructure Library (ITIL)
- Control Objectives for Information and Related Technology (COBIT)
- HIPAA – Health Insurance Portability and Accountability Act

Regulations and standards that impact physical design decisions

Three Areas of SOX Audits

- IT change management
  - Processes by which changes to operational systems and databases are authorized
- Logical access to data
  - Security procedures to prevent unauthorized access
  - Personnel controls and physical access controls
- IT operations
  - Policies and procedures for day-to-day management of infrastructure, applications, and databases

HIPAA - Health Insurance Portability and Accountability Act

- Complete Data Encryption — All health data is encrypted while in the database and during transit.
- Data Stores — If applicable, any subsystems that store encrypted BLOBs should have no 'knowledge' of what is being stored.
- Unique User IDs — HIPAA requires unique user IDs for all users and prohibits the sharing of user login credentials.
- Authentication — A database must securely authenticate users who will have access to PHI.
- Audit Logs — All data usage (user logins, reads, writes and edits) must be logged in a separate infrastructure and archived according to HIPAA requirements.
- Database Backups — Must be created, tested and securely stored. All database backups must themselves be fully encrypted. Note that, under current HIPAA Rules, data that has been properly encrypted does not trigger mandatory Breach Reporting if the data is stolen or compromised.
- Data Disposal — Methods must be in place or available to ensure that data and media are disposed of securely when no longer needed.

Physical Database Design

- Another goal of physical database design is *data processing efficiency*
- We will concentrate on choices often available to optimize performance of database services
- Physical Database Design requires information gathered during earlier stages of the design process

Physical Design Process

Inputs

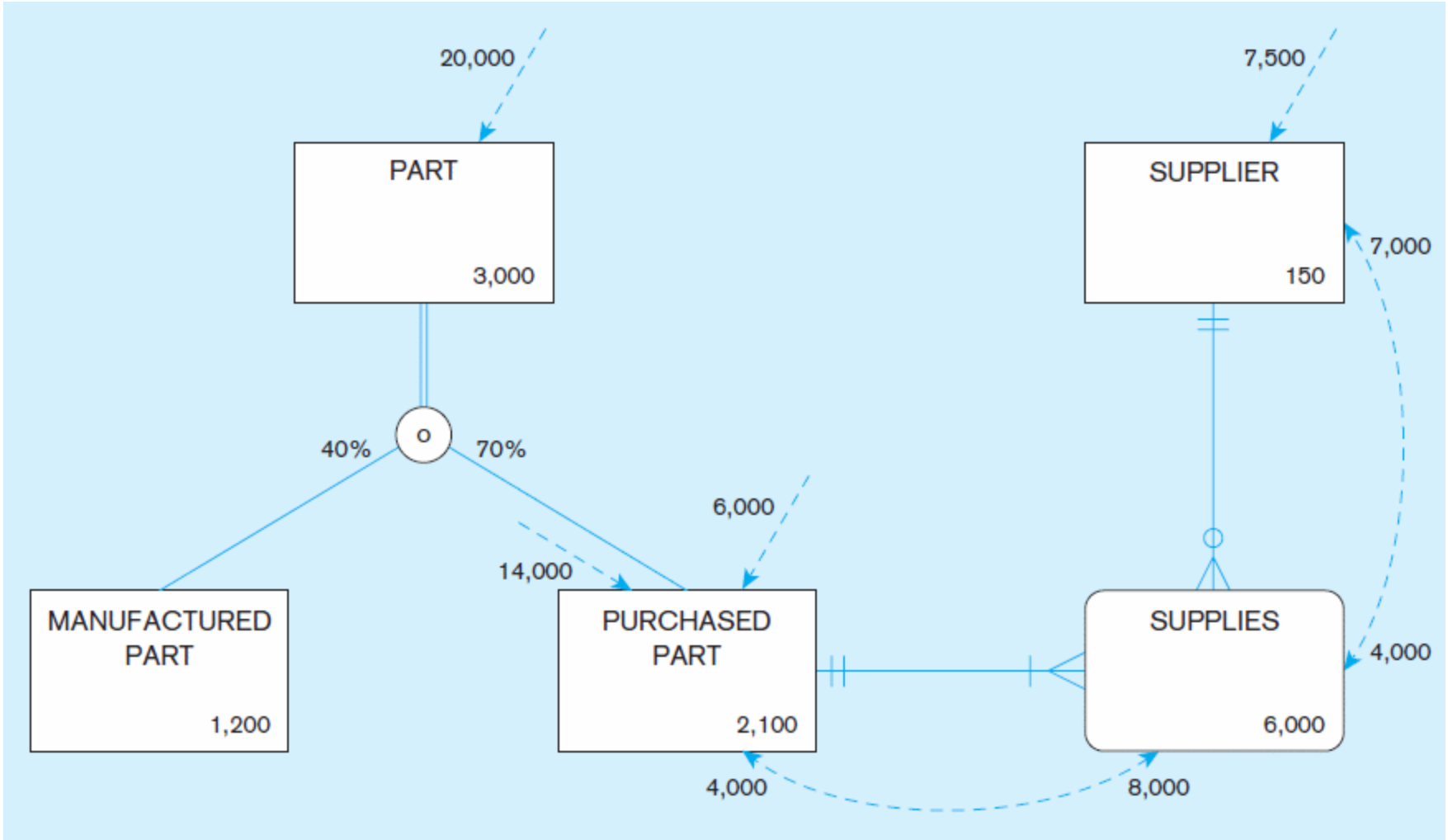
- Normalized relations
- Volume estimates
- Attribute definitions
- Response time expectations
- Data security needs
- Backup/recovery needs
- Integrity expectations
- DBMS technology used

Leads to

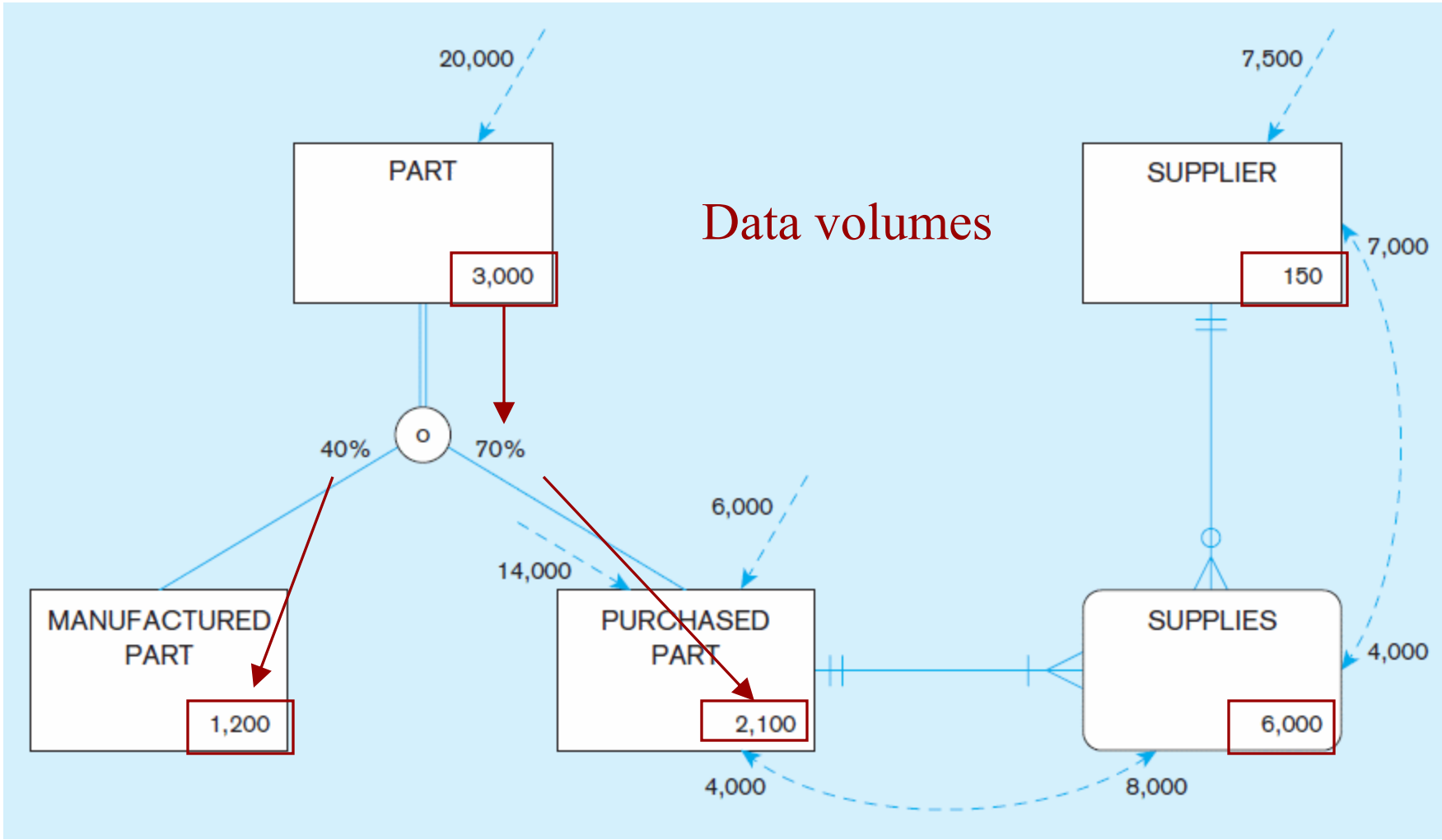
Decisions

- Attribute data types
- Physical record descriptions (doesn't always match logical design)
- File organizations
- Indexes and database architectures
- Query optimization

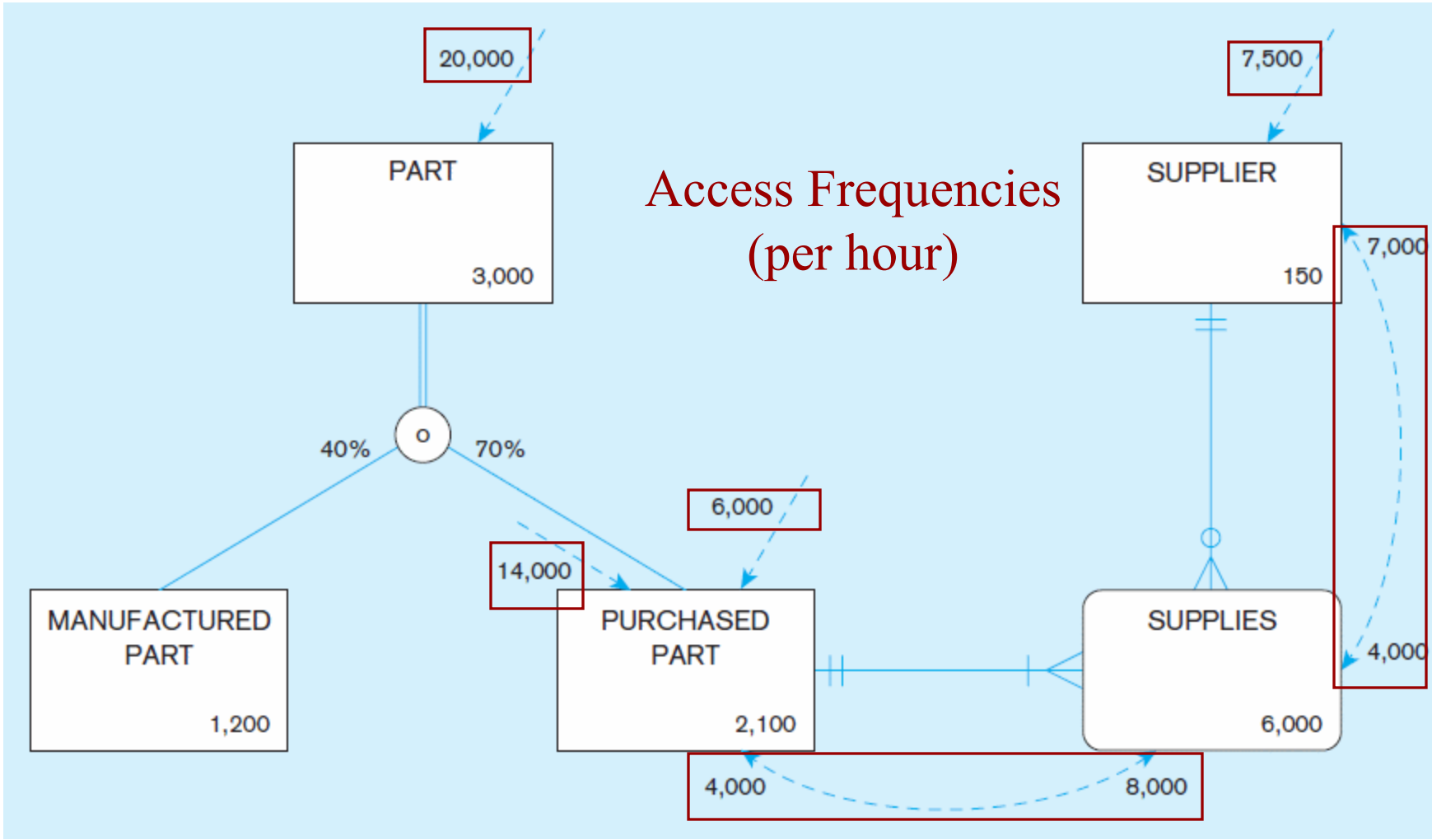
Composite usage map (Pine Valley Furniture Company)



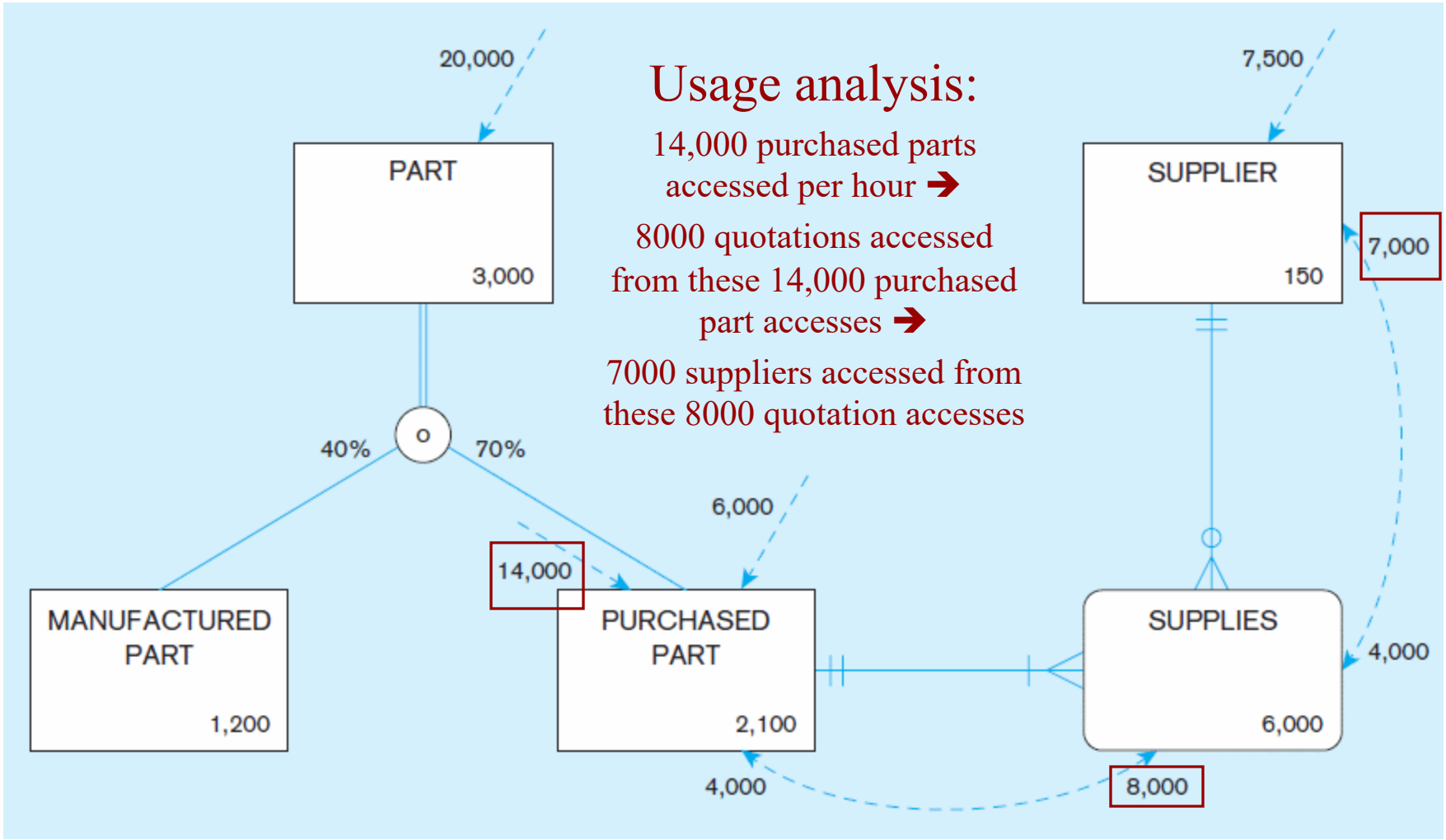
Composite usage map (Pine Valley Furniture Company) (cont.)



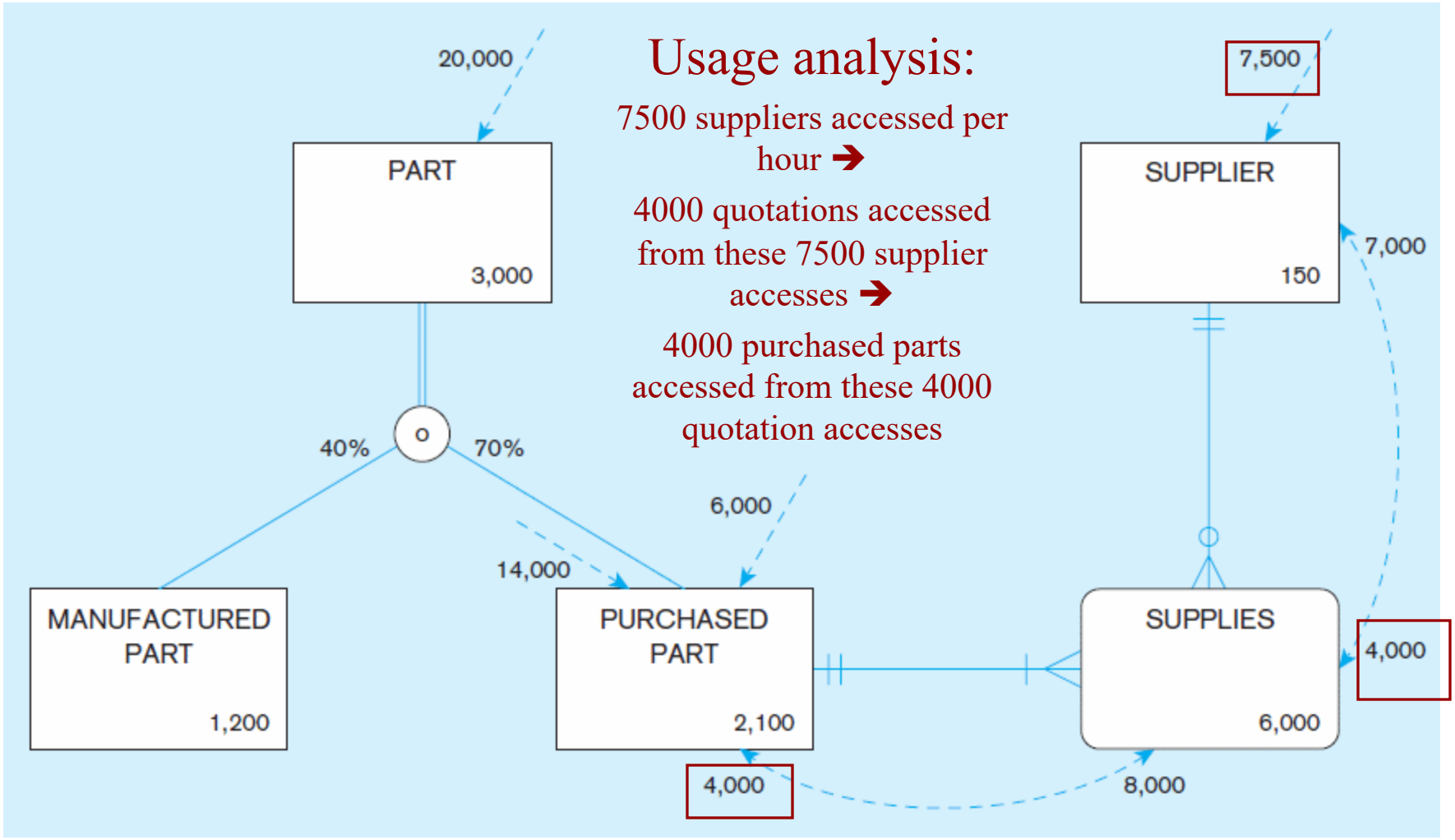
Composite usage map (Pine Valley Furniture Company) (cont.)



Composite usage map (Pine Valley Furniture Company) (cont.)



FComposite usage map (Pine Valley Furniture Company) (cont.)



Physical Design Decisions

- There are several critical decisions that will affect the integrity and performance of the system
  - Storage Format
  - Physical record composition
  - Data arrangement
  - Indexes
  - Query optimization and performance tuning

Storage Format

- Choosing the storage format of each *field* (attribute). The DBMS provides some set of data types that can be used for the physical storage of fields in the database
- Data Type (format) is chosen to minimize storage space and maximize data integrity

Objectives of data type selection

- Minimize storage space
- Represent all possible values
- Improve data integrity
- Support all data manipulations
- The correct data type **should**, in minimal space, represent every possible value (but eliminate illegal values) for the associated attribute *and* can support the required data manipulations (e.g. numerical or string operations)

Common Data Types (1 of 2)

- VARCHAR2(length) max 400 characters
  - Variable-length character data. A string that is shorter than the maximum length will consume only the required space. NVARCHAR2 is Unicode.
- CHAR(length) max 200 characters
  - Fixed length character data. NCHAR is Unicode.
- CLOB
  - Character large object, capable of storing up to 4 gigabytes of one variable length character data field
- NUMBER
  - Positive or negative number. NUMBER(5) means a 5 digit integer. NUMBER(5,2): 5 digits, two to the right of the decimal point.

Common Data Types (2 of 2)

- DATE
  - Can represent from Jan 1 4712 BC to Dec 31 9999 AD
  - Stores century, year, month, day, hour, minute, second
- TIMESTAMP
  - Like a date. Can include fractional seconds, and time zones.
- BLOB
  - Binary large object, can store up to 4 gigabytes
  - Used for photos, sound clips, etc.

Example of a code look-up table (Pine Valley Furniture Company)

PRODUCT Table

Product No	Description	Product Finish	...
B100	Chair	C	
B120	Desk	A	
M128	Table	C	
T100	Bookcase	B	
...	...	...	

PRODUCT FINISH Look-up Table

Code	Value
A	Birch
B	Maple
C	Oak

Code saves space, but costs an additional lookup to obtain actual value

Controlling Data Integrity

- Default values
- Range control
- Null value control
- Referential integrity
- Handling missing data

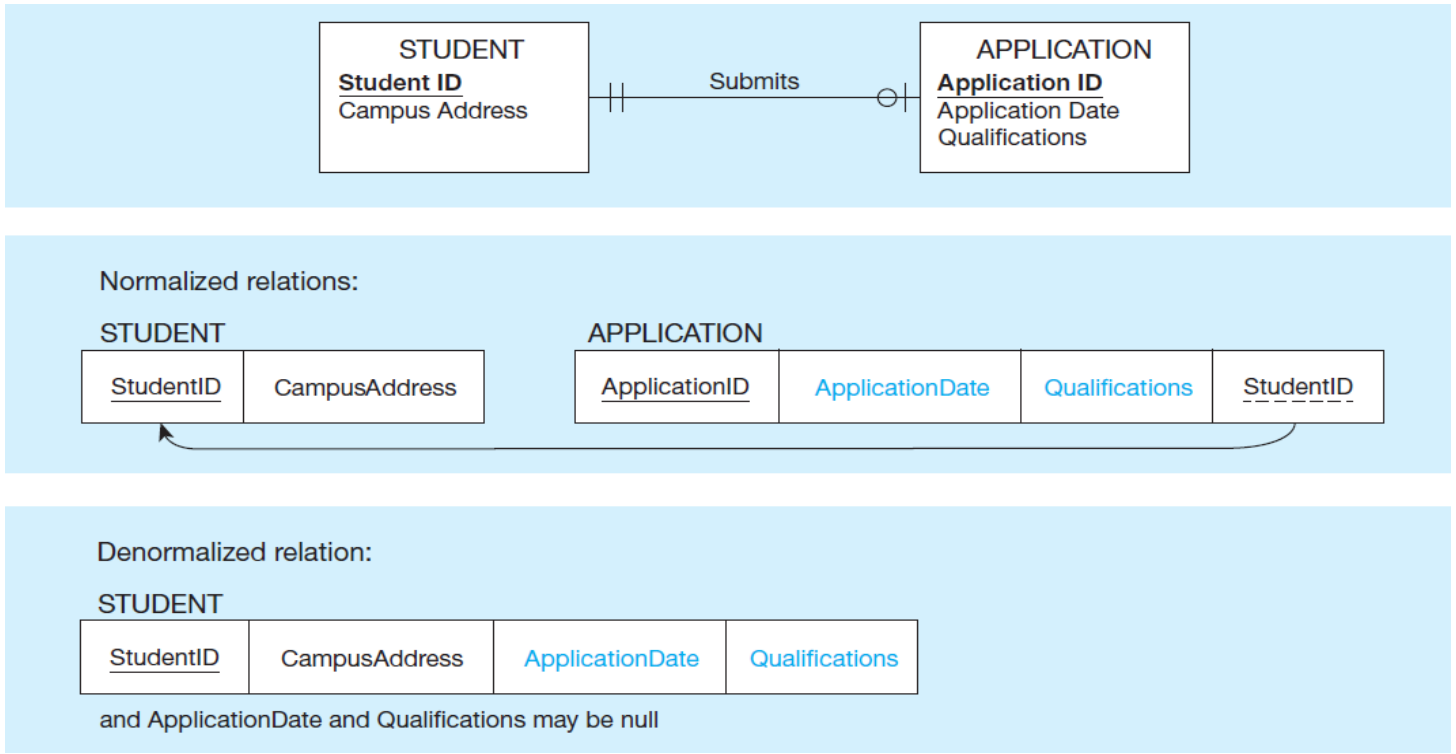
Handling Missing Data

- Substitute an estimate of the missing value (e.g., using a formula)
- Construct a report listing missing values
- In programs, ignore missing data unless the value is significant (sensitivity testing)

Denormalization

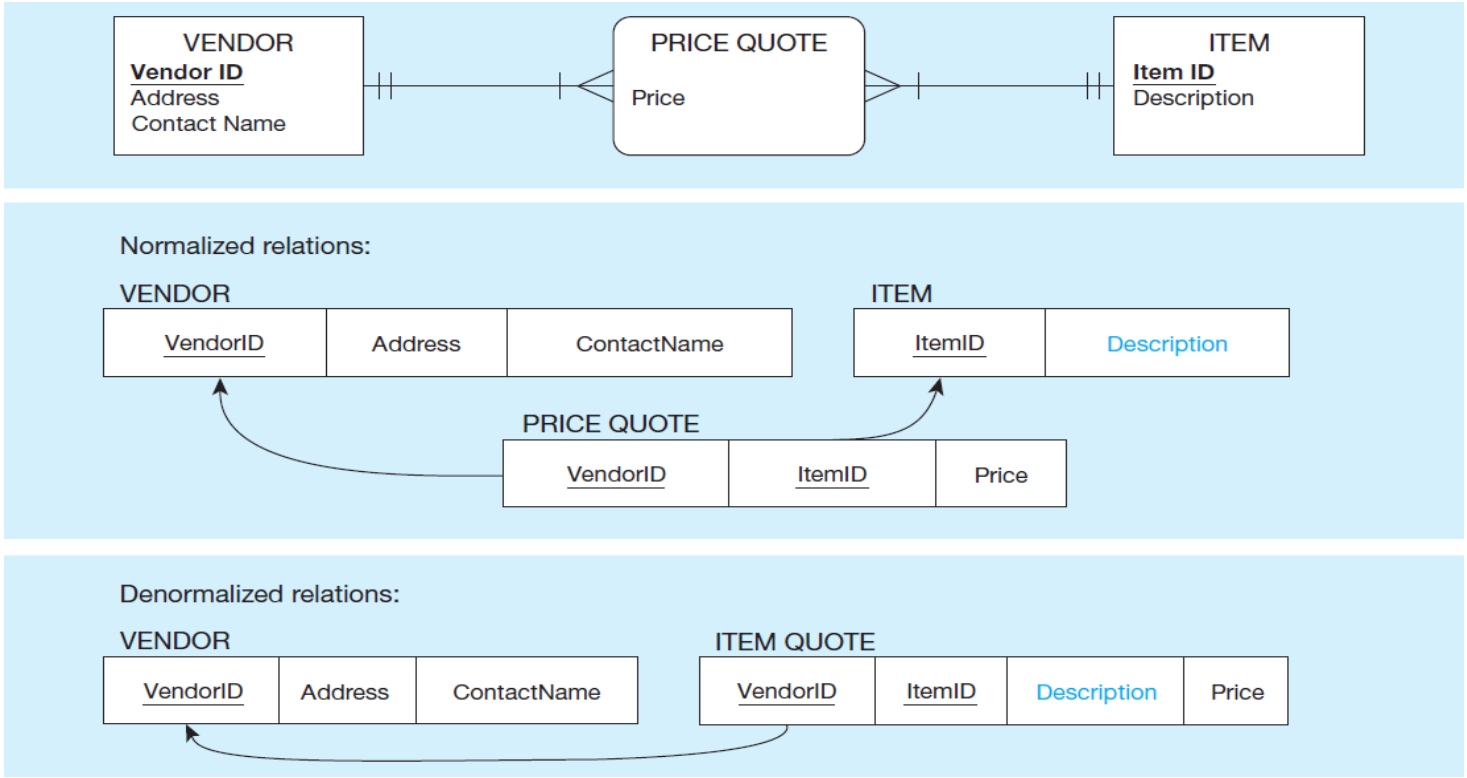
- Transforming normalized relations into non-normalized physical record specifications
- Benefits:
  - Can improve performance (speed) by reducing number of table lookups (i.e. reduce number of necessary join queries)
- Costs (due to data duplication)
  - Wasted storage space
  - Data integrity/consistency threats

A possible Denormalization Situation



Two entities with one-to-one relationship

A possible Denormalization Situation



A many-to-many relationship with non-key attributes

Denormalize with Caution

- Denormalization can
  - Increase chance of errors and inconsistencies
  - Reintroduce anomalies
  - Force reprogramming when business rules change
- Perhaps other methods could be used to improve performance of joins
  - Organization of tables in the database (file organization and clustering)
  - Proper query design and optimization

Partitioning

- Horizontal Partitioning: Distributing the rows of a logical relation into several separate tables
  - Useful for situations where different users need access to different rows
  - Three types: Key Range Partitioning, Hash Partitioning, or Composite Partitioning
- Vertical Partitioning: Distributing the columns of a logical relation into several separate physical tables
  - Useful for situations where different users need access to different columns
  - The primary key must be repeated in each file

Partitioning Pros and Cons

- Advantages of Partitioning
  - Efficiency: records used together are grouped together
  - Local optimization: each partition can be optimized for performance
  - Security: data not relevant to users are segregated
  - Recovery and uptime: smaller files take less back up time
  - Load balancing: partitions stored on different disks, reduces contention
- Disadvantages of Partitioning
  - Inconsistent access speed: slow retrievals across partitions
  - Complexity: non-transparent partitioning
  - Extra space or update time: duplicate data; access from multiple partitions

Lecture Outline

- Physical Database Design
- Access Methods and the Cloud

Authentication Schemes (1 of 2)

- Goal – obtain a **positive** identification of the user
- Passwords: First line of defense
  - Should be at least 8 characters long
  - Should combine alphabetic and numeric data
  - Should not be complete words or personal information
  - Should be changed frequently

Authentication Schemes (2 of 2)

- Strong Authentication
  - Passwords are flawed:
    - Users share them with each other
    - They get written down, could be copied
    - Automatic logon scripts remove need to explicitly type them in
    - Unencrypted passwords travel the Internet
- Possible solutions:
  - Two factor – e.g., smart card plus PIN
  - Three factor – e.g., smart card, biometric, PIN

Cloud-based Data Management Services (1 of 2)

- Cloud computing
  - Provisioning/acquiring computing services on demand using centralized resources accessed through public Internet or private networks
- Infrastructure-as-a-Service (IaaS)
  - Cloud service involving hardware and various types of systems software resources
- Platform-as-a-Service (PaaS)
  - Cloud service involving hardware and various types of systems software resources

Cloud-based Data Management Services
(2 of 2)

- Software-as-a-Service (SaaS)
  - Cloud service involving software solutions/applications intended to directly address the needs of a noncomputing activity
- Database-as-a-Service (DBaaS)
  - Cloud service involving data management cloud platform service

Benefits of Cloud-based Data Management Services

- No need for initial investments in hardware, physical facilities, and systems software
- Significantly lower need for internal expertise in the management of the database infrastructure
- Better visibility of overall costs of data management
- Increased level of flexibility (elasticity) in situations when capacity needs to fluctuate significantly
- Allows organizations to explore new data management technologies more easily
- Mature cloud service providers have expertise to provide a high level of availability, reliability, and security

Downside of Cloud-based Data Management Services

- Existing systems do not yet provide capacity using a model that would automatically adapt to the changing requirements targeting the system
- Current systems are not yet providing full consistency guarantees in a highly distributed environment
- Live migration is still a challenging task that requires manual planning, initiation, and management
- It is challenging to be able to monitor the extent to which cloud providers are maintaining their Service Level Agreement (SLA) commitments
- DBaaS solutions are still struggling to find fully scalable models for providing ACID support for transactions