

Chapter 12

Application Architecture and Modeling

Objectives

- Define an information system's architecture in terms of the KNOWLEDGE, PROCESSES, and COMMUNICATION building blocks.
- Differentiate between logical and physical data flow diagrams, and explain how physical data flow diagrams are used to model an information system's architecture.
- Describe centralized and distributed computing alternatives, including client/server and Internet-based computing options.
- Describe database and data distribution alternatives for system design.
- Describe user and system interface alternatives for system design.
- Describe various software development environments for information system design.
- Describe strategies for developing or determining architecture of an information system.
- Draw physical data flow diagrams for an information system's architecture and processes.

Application Architecture

Application architecture – a specification of the technologies to be used to implement information systems. The blueprint to communicate the following design decisions:

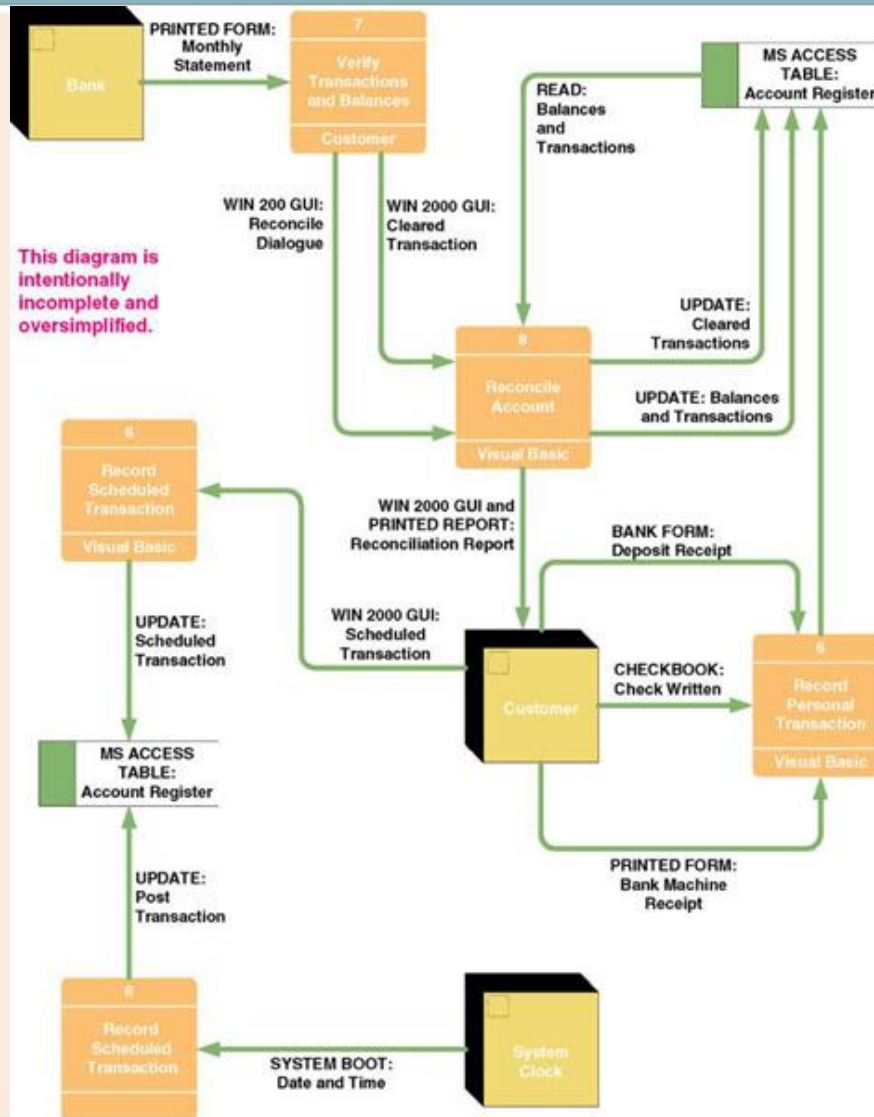
- The degree to which the information system will be centralized or distributed.
- The distribution of stored data.
- The implementation technology for software developed in-house.
- The integration of commercial off-the-shelf software.
- The technology to be used to implement the user interface.
- The technology to be used to interface with other systems

Physical Data Flow Diagram (DFD)

Physical data flow diagram (DFDs) – a process model used to communicate the technical implementation characteristics of an information system.

- Communicate technical choices and other design decisions to those who will actually construct and implement the system.
- Recall from Chapter 9 that DFDs are a type of process model.

Sample Physical Data Flow Diagram



Physical Processes

Physical process – either a *processor*, such as a computer or person, or a technical implementation of specific work to be performed, such as a computer program or manual process.

- Logical processes may be assigned to physical processors such as PCs, servers, people, or devices in a network. A physical DFD would model that network structure.
- Each logical process requires an implementation as one or more physical processes.
- A logical process may be split into multiple physical processes:
 - To define aspects performed by people or computers.
 - To define aspects implemented by different technologies.
 - To show multiple implementations of the same process.
 - To add processes for exceptions and security.

Physical Process Notation

ID (optional)




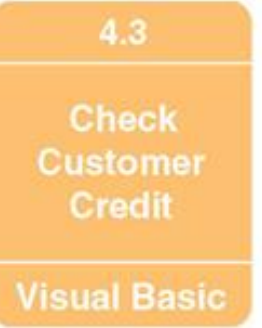

Action Verb

+

**Noun or Object
Phrase**

Implementation

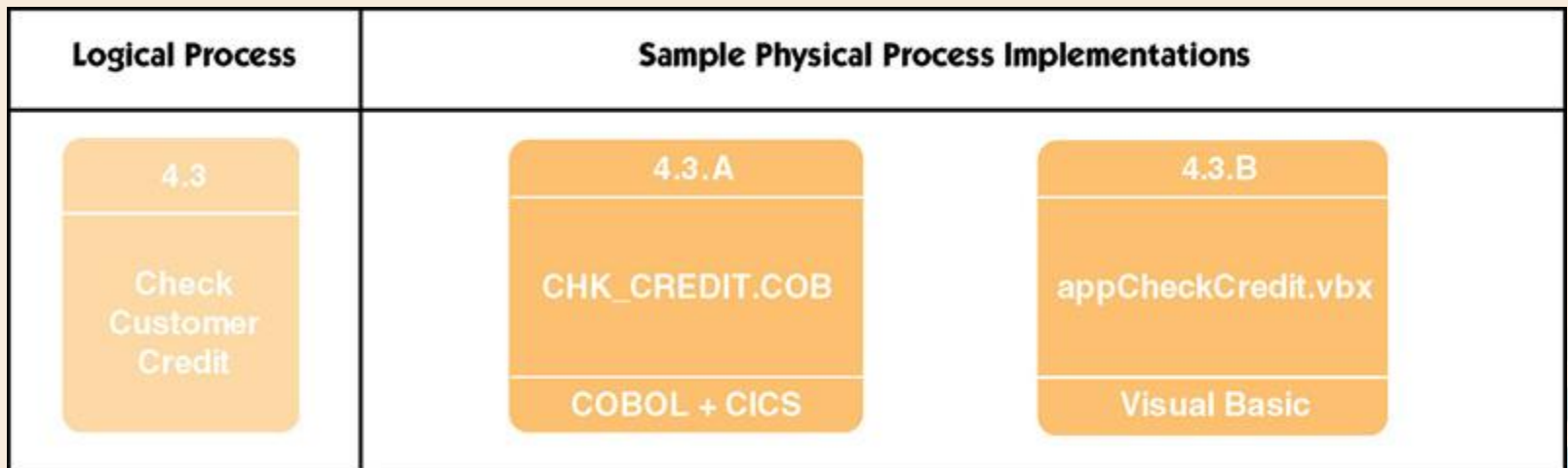
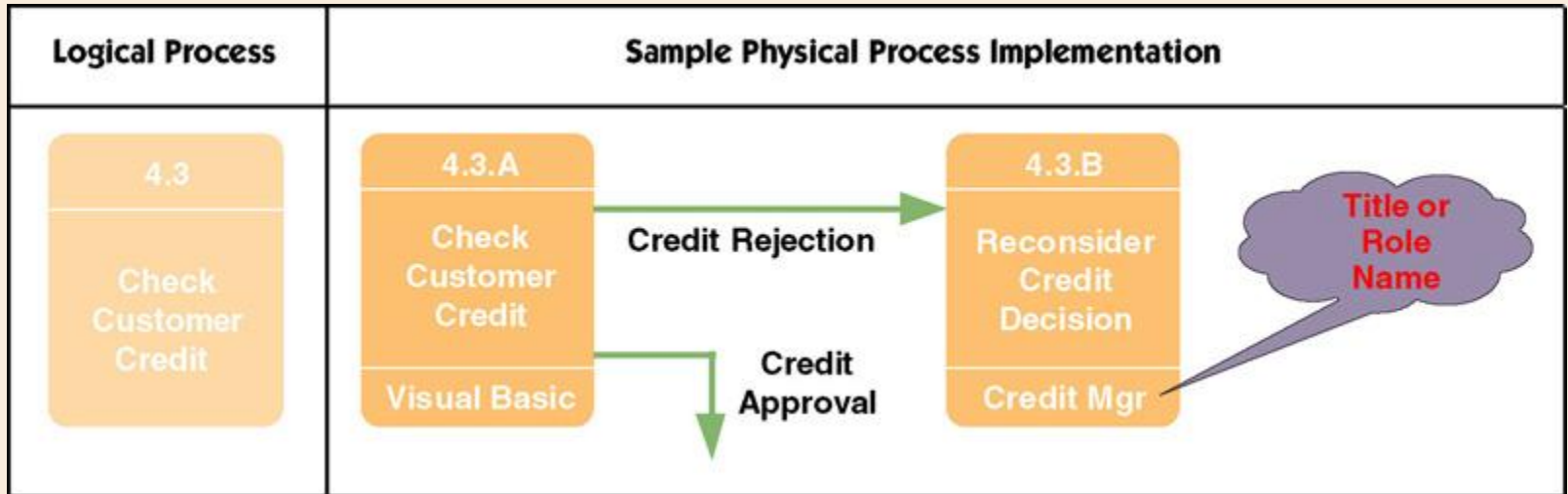
Samples of Physical Processes

Logical Process	Sample Physical Process Implementations			
 <p>4.3 Check Customer Credit</p>	 <p>4.3 Check Customer Credit Acct Clerk</p>	 <p>4.3 Check Customer Credit COBOL/CICS</p>	 <p>4.3 Check Customer Credit Visual Basic</p>	 <p>4.3 Check Customer Credit Quickbooks</p>

Possible Computer Process Implementations

- A purchased application software package
 - Also called *commercial off-the-shelf (COTS) software*
- A system or utility program
 - Such as an e-mail/message server or third-party framework
- An existing application program from a program library
 - May require modification
- A program to be written

Sample Physical Process Implementations



Physical Data Flows

A physical data flow represents:

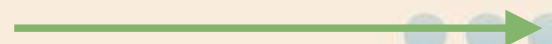
- Planned implementation of an input to, or output from a physical process.
- Database command or action such as create, read, update, or delete.
- Import of data from, or export of data to another information system.
- Flow of data between two modules or subroutines (represented as physical processes).

Implementation method:
Data flow name

















OR



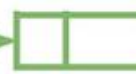


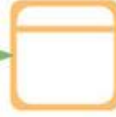
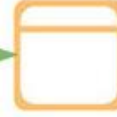






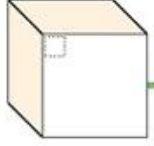

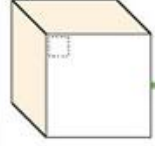
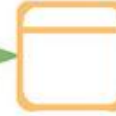
Data flow name
(Implementation method)



Sample Physical Data Flows

Logical Data Flow	Implementation	Sample Physical Data Flow
Order → 	Computer Input (Keyboard)	WIN 2000 GUI: Order Form → 
Order → 	Computer Input (Internet)	HTML: Order Form → 
Product Sold → 	Computer Input (Keyless)	BAR CODE: Product UPC → 
Hours Worked → 	Computer Input (Batch File)	KEY-TO-DISK: Hours Worked → 
 Salary Equity Analysis →	Computer Output (Printed)	 PRINTOUT: Salary Equity Report →
 Account History →	Computer Output (On-Line)	 WIN 2000 GUI: Account History →
→ Create Order → 	Create a record in a database	→ SQL Insert: New Order → 

Sample Physical Data Flows (continued)

 Unfilled Orders →	Read records in a database	SQL Select: Unfilled Orders →
Update Credit rating → 	Update a record in a database	SQL Update: Credit Rating → 
Delete Employee → 	Delete a record in a database	SQL Delete: Employee → 
Insurance Accident Claim → 	Import a data file	IMAGE FILE: Insurance Accident Claim → 
 Schedule of Classes →	Export a data file	 Comma Delimited File: Schedule of Classes →
 Extended Cost → 	Pass data between modules of a program	 Extended Cost → 
 Course Request → 	Pass a manual form	 Form 23: Course Request → 

Physical External Agents

Physical external agents are carried over from the logical DFD models.

- If scope changes, the logical models should be changed before the physical models are drawn.

Physical Data Stores

A physical data store represents the planned implementation of one of:

- A database
- A table in a database
- A computer file
- A tape or media backup of anything important
- A temporary file or batch
- Any type of noncomputerized file

Physical Data Store Notation



Physical Data Store Implementations

Logical Data Store	Implementation	Physical Data Store
 Human Resources	A database (multiple tables)	 Oracle : Human Resources DB
 Marketing	A database view (subset of a database)	 SQL Server: Northeast Marketing DB
 Purchase Orders	A table in a database	 MS Access: Purchase Orders
 Accounts Receivable	A legacy file	 VSAM File: Accounts Receivable
 Tax Rates	Static data	 ARRAY: Tax Table
 Orders	An off-line archive	 TAPE Backup: Closed Orders
 Employees	A file of paper records	 File Cabinet: Personnel Records
 Faculty/Staff Contact Data	A directory	 Handbook: Faculty/Staff Directory
 Course Enrollments By Date	Archived reports (for reuse and recall)	 REPORT MGR: Course Enrollment Reports

Distributed versus Centralized Systems

Distributed system – a system in which components are distributed across multiple locations and computer networks.

- Accordingly, the processing workload is distributed across multiple computers on the network.

Centralized systems – a system in which all components are hosted by a central, multi-user computer.

- Users interact with the system via terminals (or a PC emulating a terminal).
- Virtually all the actual processing and work is done on the host computer.

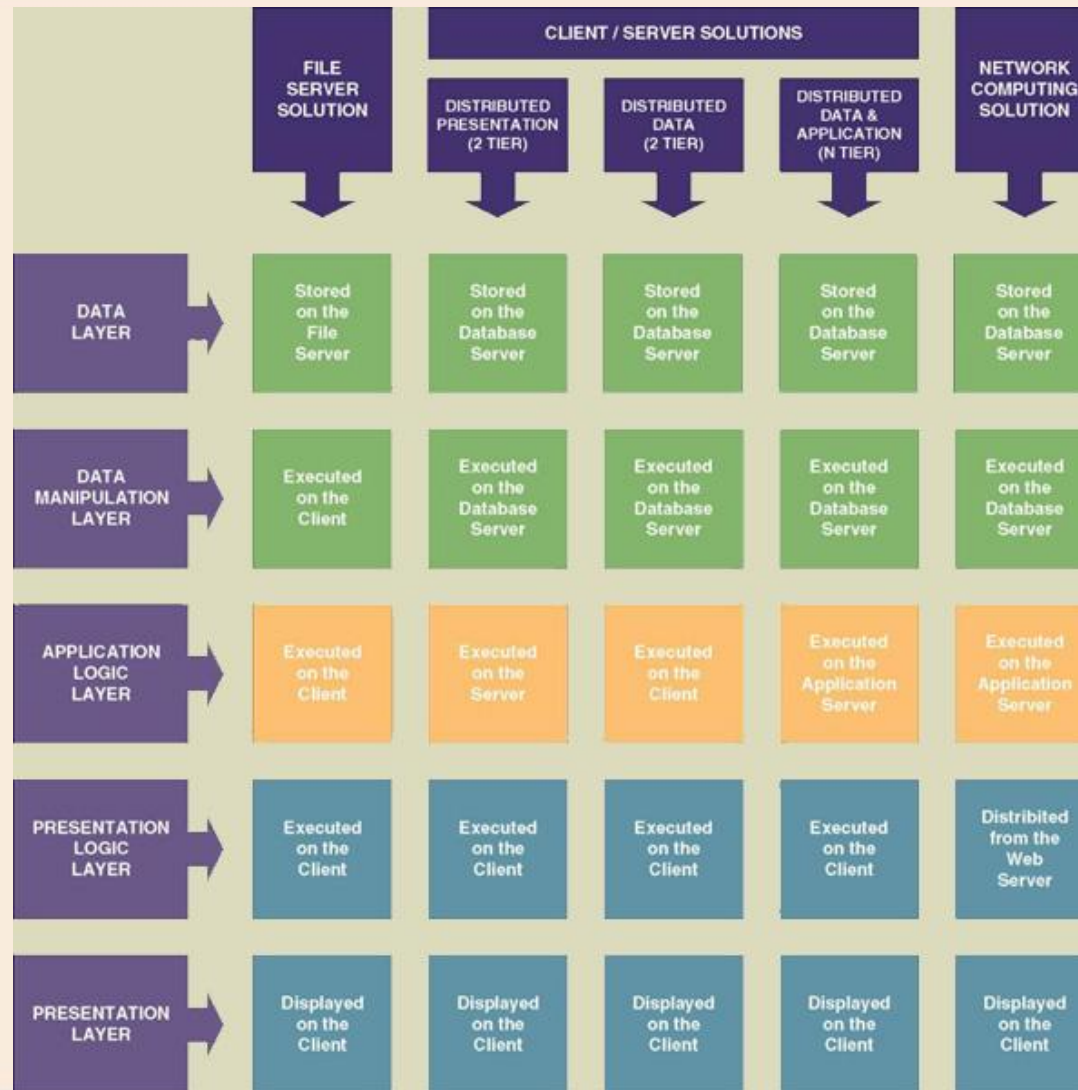
Why the Trend Toward Distributed Systems?

- Modern businesses are already decentralized (distributed).
- Distributed computing moves information and services closer to the customers and users who need them.
- Distributed computing consolidates the power of personal computers across the enterprise.
- Distributed computing solutions are in general more user-friendly because they use the PC as the user interface processor.
- Personal computers and network servers are less expensive than mainframe computers
 - Though total cost of ownership is at least as expensive

Computing Layers

- **Presentation layer**—the user interface
- **Presentation logic layer**—processing that must be done to generate the presentation, such as editing input data or formatting output data.
- **Application logic layer**—the logic and processing to support business rules, policies, and procedures
- **Data manipulation layer**—to store and retrieve data to and from the database
- **Data layer**—the actual business data

Types of Distributed Computing



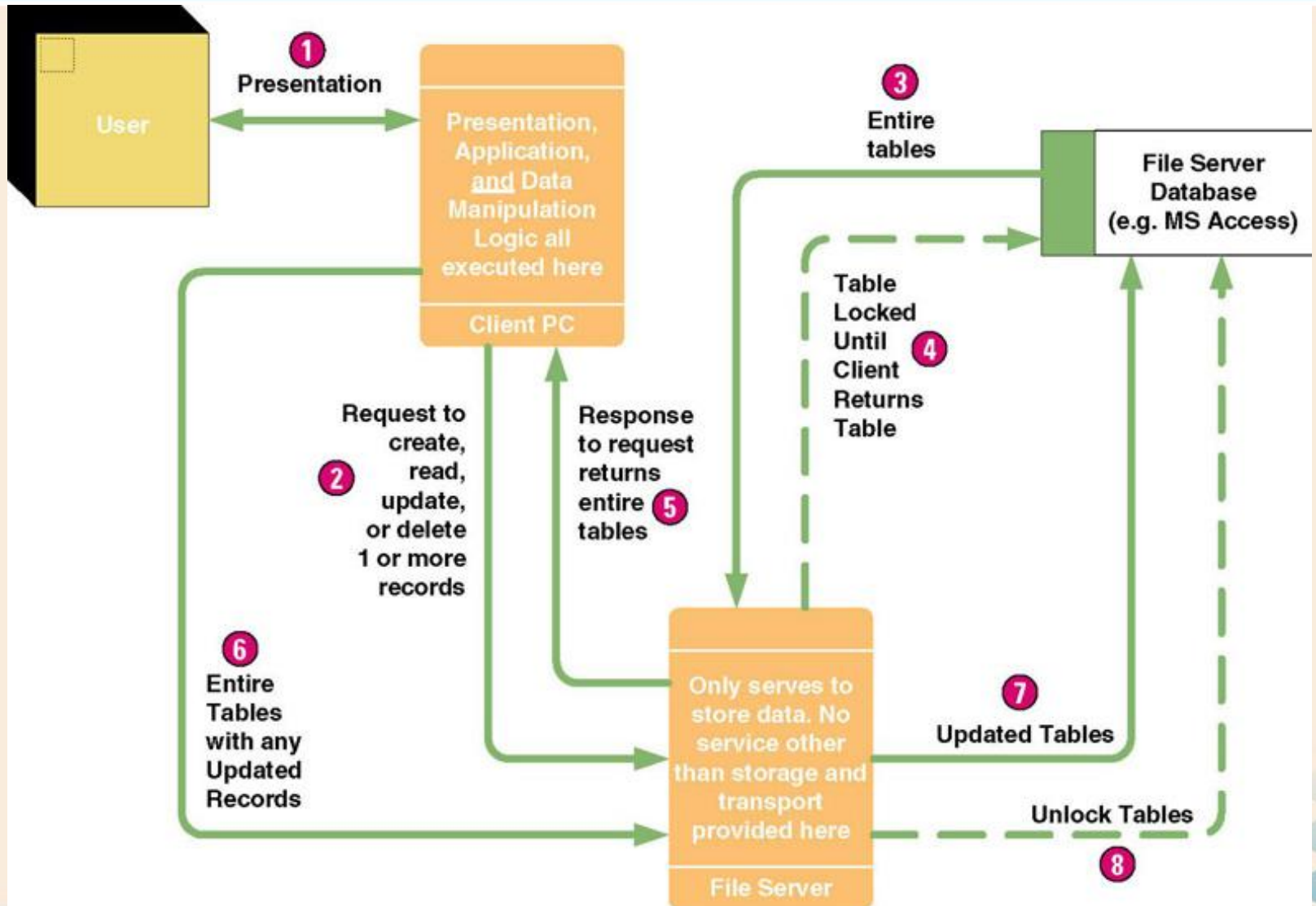
File Server Architecture

Local area network (LAN) – a set of client computers (PCs) connected over a relatively short distance to one or more servers.

File server system – a LAN in which a server hosts the data of an information system.

- All other layers are implemented on the client computers.
- Frequently excessive network traffic to transport data between servers and clients.
- Client must be fairly robust (“fat”) because it does most of the work.
- Database integrity can be compromised.

File Server Architecture



Client/Server Architecture — Clients

Client/server system – a distributed computing solution in which the presentation, presentation logic, application logic, data manipulation, and data layers are distributed between client PCs and one or more servers.

Thin client – a personal computer that does not have to be very powerful because it only presents the user interface to the user.

Fat client – a personal computer, notebook computer, or workstation that is typically powerful.

Client/Server Architecture — Servers

- **Database server** – a server that hosts one or more databases and executes all data manipulation commands at the server.
- **Transaction server** – a server that hosts services which ensure that all database updates for a transaction succeed or fail as a whole.
- **Application server** – a server that hosts application logic and services for an information system.
- **Messaging or groupware server** – a server that hosts services for e-mail, calendaring, and other work group functionality.
- **Web server** – a server that hosts Internet or intranet websites.

Client/Server—Distributed Presentation

Distributed presentation – a client/server system in which the presentation and presentation logic layers are shifted from the server to reside on the client.

- The application logic, data manipulation, and data layers remain on the server (frequently a mainframe).
- Character user interface (CUI)
- Graphical user interface (GUI)

Building a GUI From a CUI – Screen Scrapers

```
KP11-----Kindle Press - Confidential -----TRXZ2356
Display Title Information

ISBN: 0-767-54221-5      Qty in Stock: 865
Title: So You Want to   Qty Consigned: 234
List Price: $31.95      Cost Each: $5.41
Author Code: PTC

Status: T
Group: 7-

Reorder Level: 215

Carton Count: 12
Ctn Wt (lbs): 22.6
```

Online help

Information from database

Information from host application

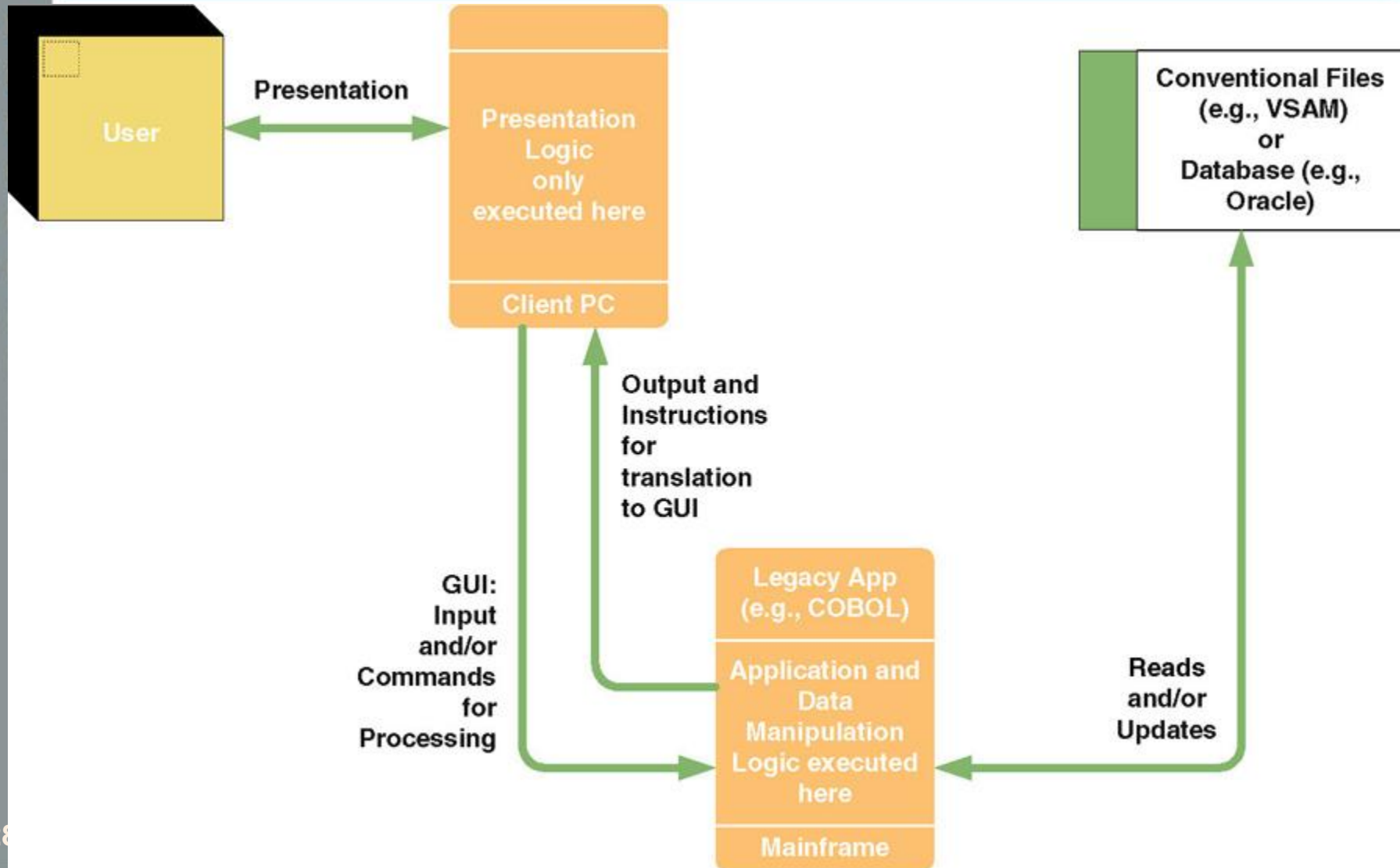
Runs over the Internet

Runs over the Internet

Image from database

Custom graphical buttons

Client/Server—Distributed Presentation

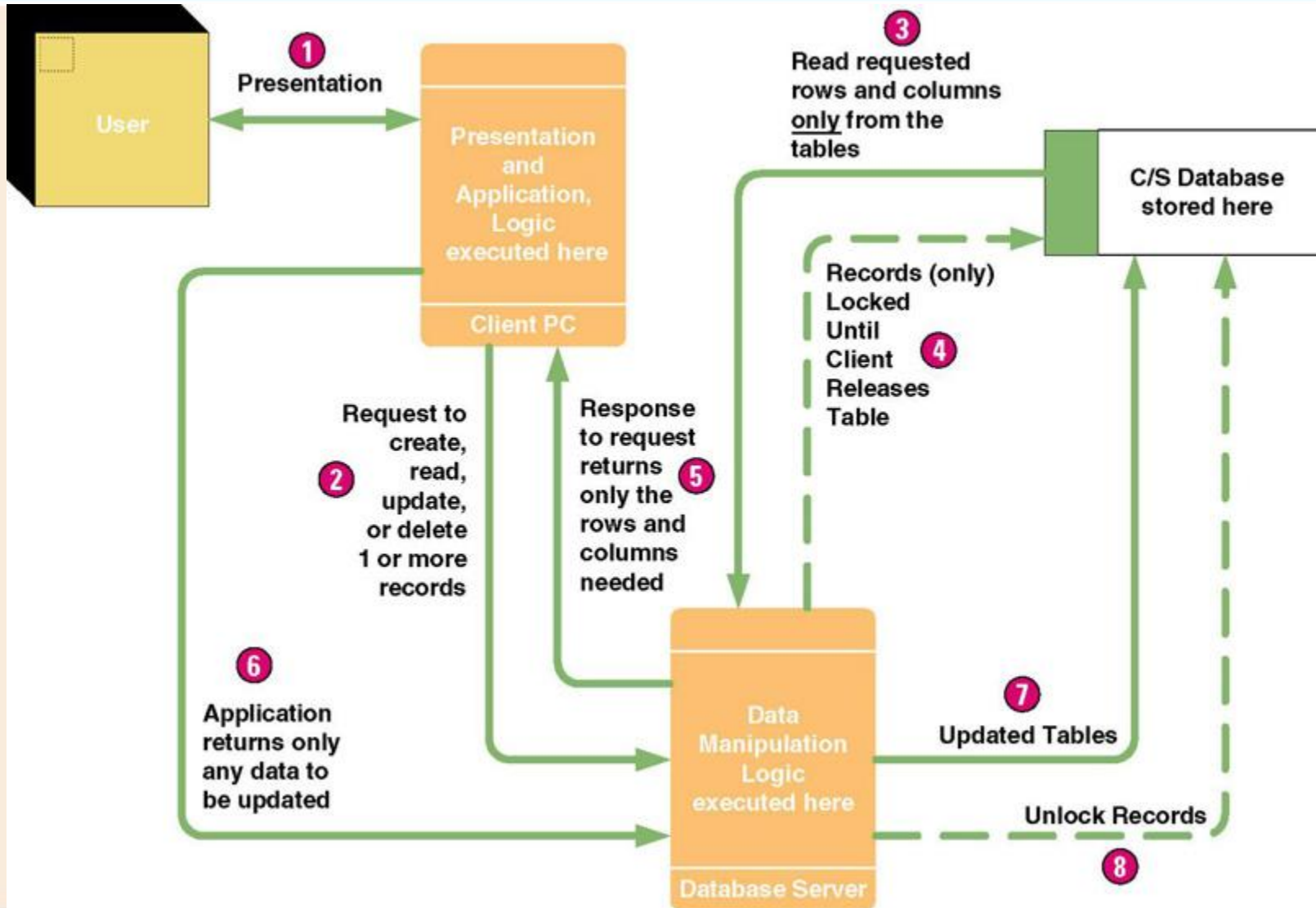


Client/Server—Distributed Data

Distributed data – a client/server system in which the data and data manipulation layers are placed on the server(s), and other layers are placed on the clients.

- Sometimes called two-tiered client/server computing.
- Difference to file server systems is where the data manipulation commands are executed.
- Much less network traffic than file server systems because only the database requests and the results of those requests are transported across the network.
- Database integrity is easier to maintain.

Client/Server—Distributed Data



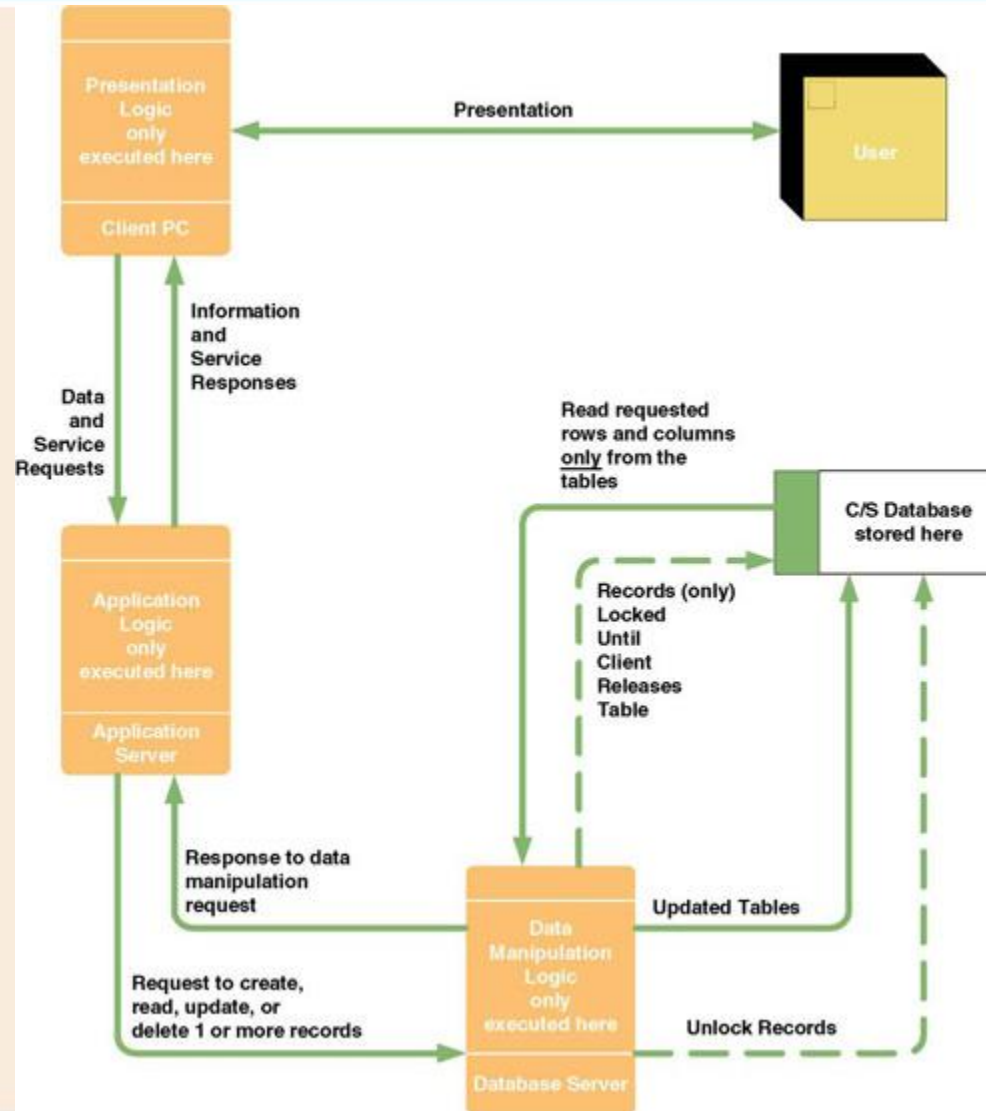
Client/Server—Distributed Data and Application

Distributed data and application – client/server system:

1. The data and data manipulation layers are placed on their own server(s),
 2. The application logic is placed on its own server,
 3. The presentation logic and presentation layers are placed on the clients.
- Also called **three-tiered** or **n-tiered client/server computing**.
 - Requires design partitioning.

Partitioning – the art of determining how to best distribute (duplicate) application components across the network.

Client/Server — Distributed Data and Application



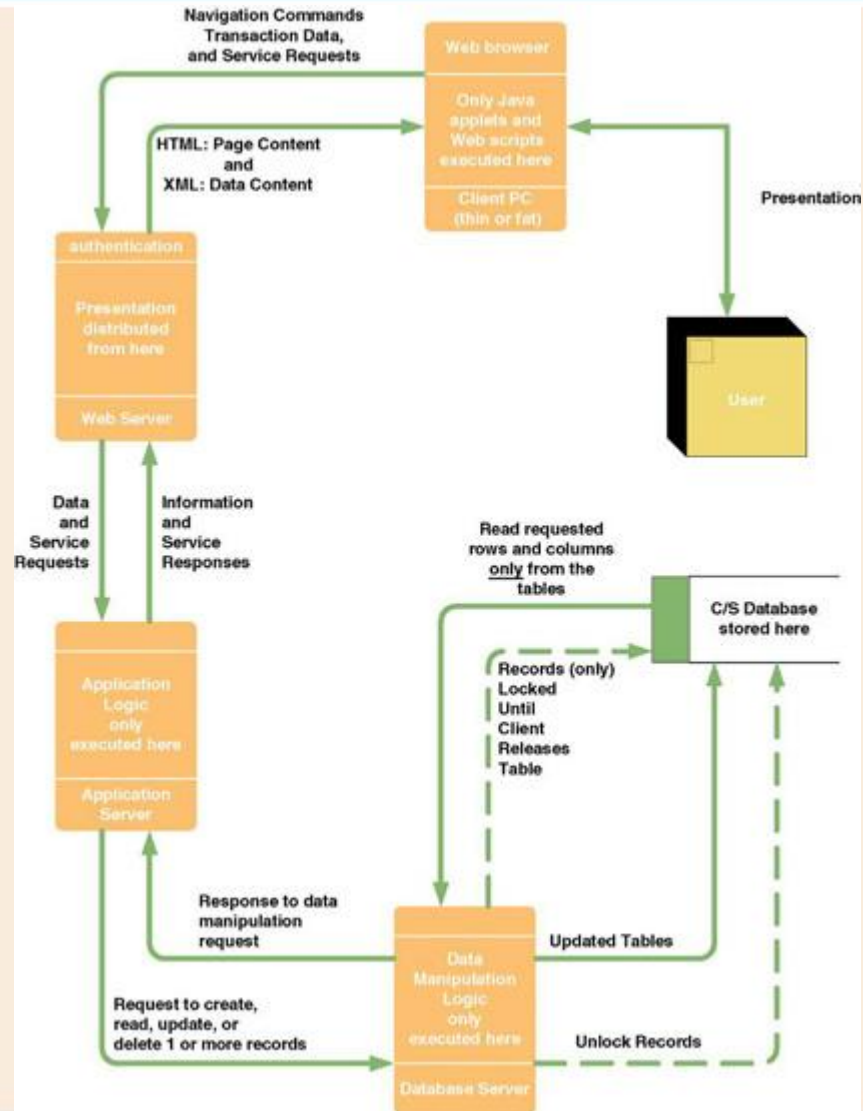
Internet- and Intranet-based Architectures

Network computing system – presentation and presentation logic layers implemented in client-side Web browsers using content downloaded from a Web server.

- Presentation logic layer connects to application logic layer running on application server, which connects to database servers on the backside of the system.
- The greatest potential of this approach is its applicability to redesign of traditional information systems to run on an intranet.

Intranet – a secure network that uses Internet technology to integrate desktop, work group, and enterprise computing into a cohesive framework.

Network Computing System: Internet/Intranet



Internet and Intranet Technologies

- Java
 - Mostly for programming server-side application logic called “servlets”
 - Occasionally for programming client-side application logic called “applets”
- HTML (HyperText Markup Language)
 - Mostly for programming the presentation layer
- XML (Extensible Markup Language)
 - Mostly for programming data content to be transported across the web
- SQL (Structured Query Language)
 - Universal standard language for database manipulation
- Web Browsers

Data Architectures

Relational database stores data in tabular form. Each file is implemented as a table. Each field is a column in the table. Related records between two tables are implemented by intentionally duplicated columns in the two tables.

Distributed relational database – A database system that duplicates tables to multiple database servers located in geographically important locations.

Distributed relational database management system – a software program that controls access to and maintenance of stored data in the relational format.

Types of Data(base) Distribution

Data partitioning truly distributes rows and columns of tables to specific database servers with little or no duplication between servers.

- Vertical partitioning assigns different columns to different servers.
- Horizontal partitioning assigns different rows to different servers.

Data replication duplicates some or all tables on more than one database server.

- Propagates updates on one database server to any other database server where the data is duplicated.

Data Partitioning versus Data Replication

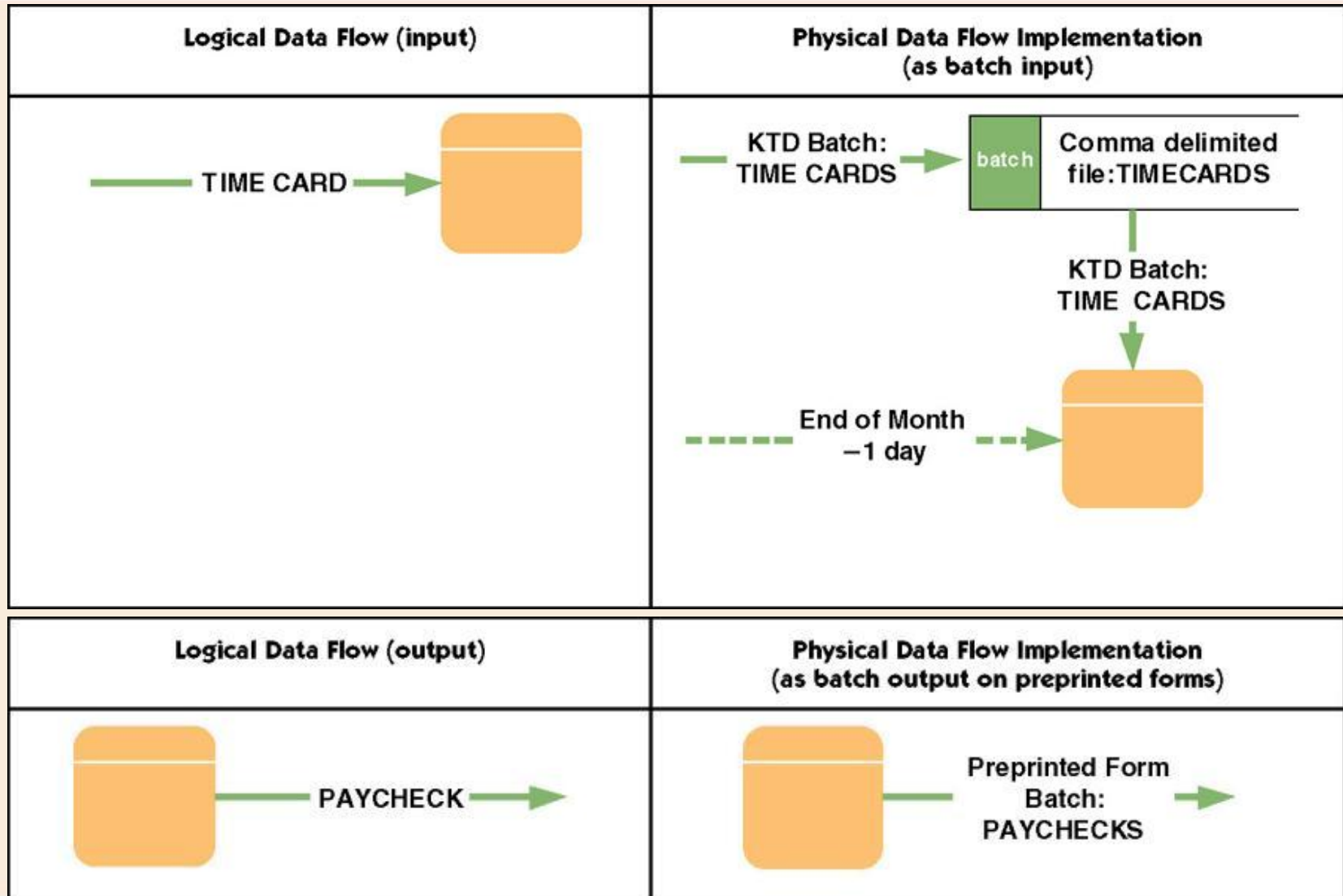
Logical Data Store	Physical Data Stores Using Partitioning	Physical Data Stores Using Replication
<div data-bbox="247 461 369 611">1</div> <div data-bbox="421 505 683 544">CUSTOMERS</div>	<div data-bbox="826 461 948 611">1P.#</div> <div data-bbox="1006 468 1263 596">Oracle 7: REGION 1 CUSTOMERS</div> <hr/> <div data-bbox="826 654 948 803">1P.#</div> <div data-bbox="1006 661 1263 789">Oracle 7: REGION 2 CUSTOMERS</div>	<p>Not applicable. Branch offices do not need access to data about customers outside of their own sales region.</p>
<div data-bbox="247 886 369 1036">2</div> <div data-bbox="436 931 668 969">PRODUCTS</div>	<p>Not applicable. All branch offices need access to data for all products, regardless of sales region.</p>	<div data-bbox="1406 886 1528 1036">2M</div> <div data-bbox="1599 893 1831 1022">Oracle 8i: PRODUCTS (Master)</div> <hr/> <div data-bbox="1406 1079 1528 1229">2R</div> <div data-bbox="1541 1086 1889 1215">Oracle 8i: PRODUCTS (Replicated Copy)</div>







Interface Architectures – Inputs, Outputs, & Middleware

- Batch inputs and outputs
- Online inputs and outputs
- Remote batch
- Keyless data entry (and automatic identification)
- Pen input
- Electronic messaging and work group technology
- Electronic Data Interchange (EDI)
- Imaging and document interchange
- Middleware

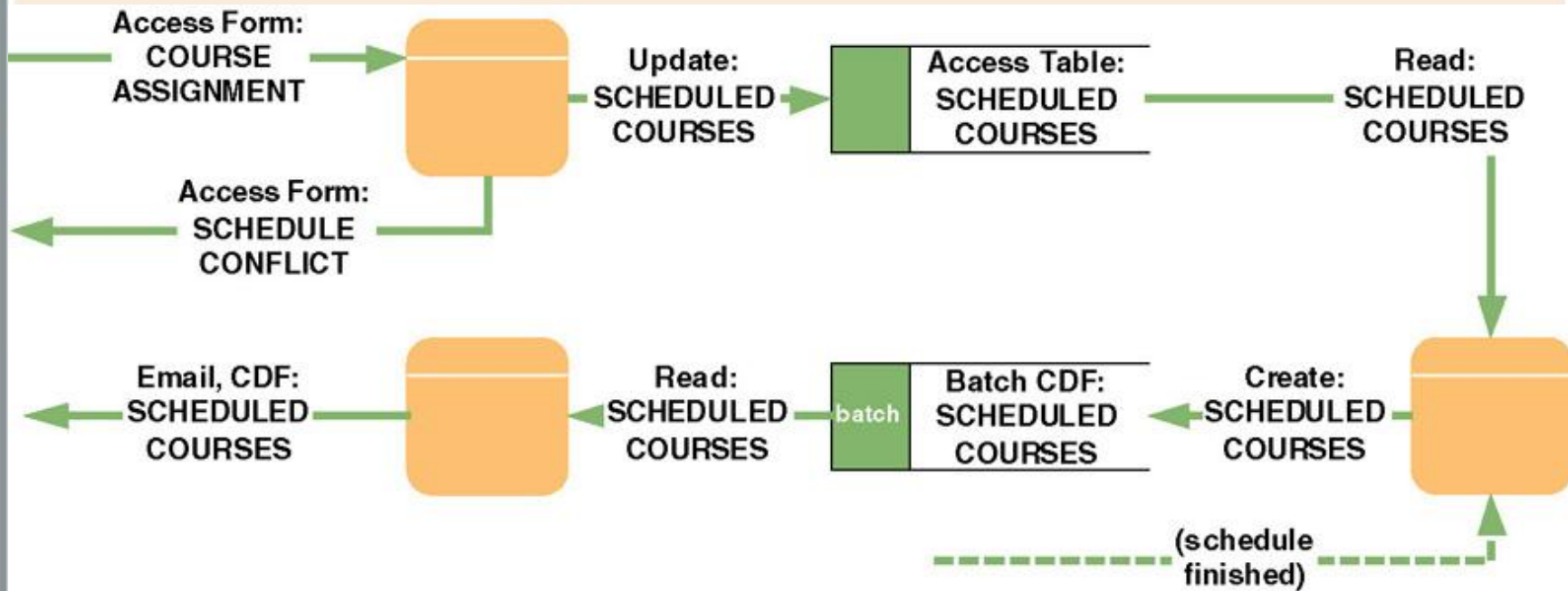
Batch Inputs and Outputs







On-Line Inputs and Outputs

Logical Data Flow (input and output)	Physical Data Flow Implementation (as on-line input and output; 2 alternatives shown)
 <p>INSURANCE CLAIM</p>	 <p>Win 2000 Form: INSURANCE CLAIM</p>
 <p>ORDER CONFIRMATION</p>	 <p>HTML Form: ORDER CONFIRMATION</p> <p>MAPI Email Message: ORDER CONFIRMATION</p>

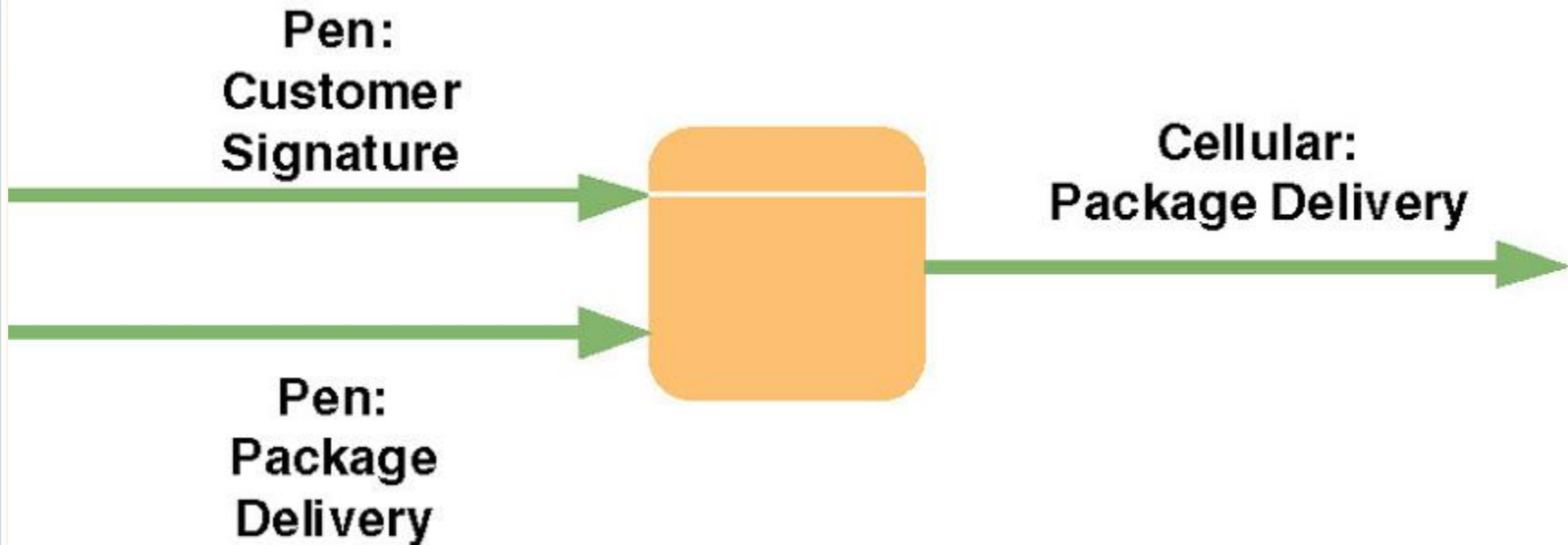
Remote Batch



Keyless Data Entry (and Automatic Identification)

Logical Data Flow (input)	Physical Data Flow Implementation (optimal mark form batch input)
	
Logical Data Flow (input)	Physical Data Flow Implementation (automatic ID input)
	

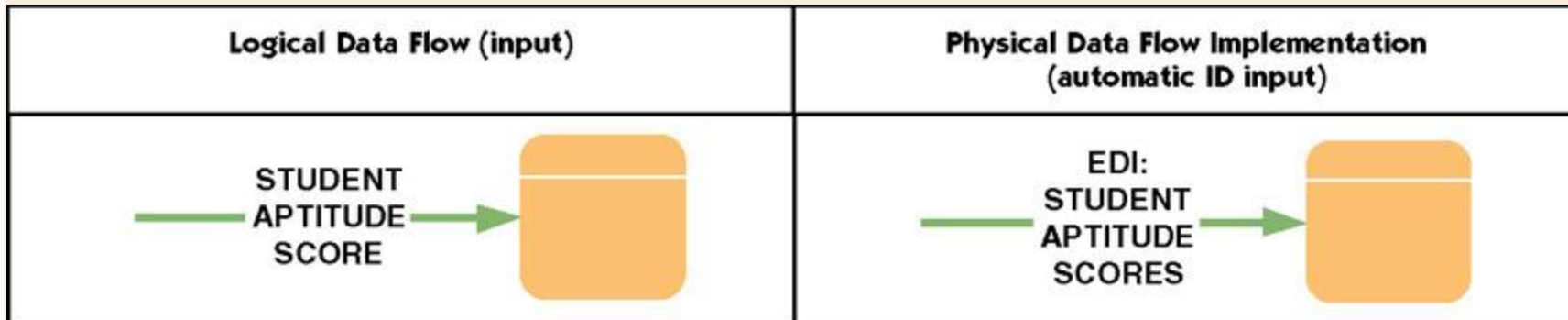
Pen Input



Electronic Data Interchange (EDI)

Electronic Data Interchange (EDI) – the standardized electronic flow of business transactions or data between businesses.

- Typically, many businesses must agree to a common data format to make EDI feasible.



Middleware

Middleware – utility software that enables communication between different processors in a system.

- It may be built into the respective operating systems or added through purchased middleware products.
- Presentation middleware
- Application middleware
- Database middleware

Process Architectures

Software development environment (SDE) – a language and tool kit for developing applications.

- SDEs exist for centralized computing
- SDEs exist for distributed presentation
- SDEs exist for two-tiered client/server
- SDEs exist for multi-tiered client/server
- SDEs exist for Internet and intranet client/server

Clean layering – a design strategy that requires that presentation, application, and data layers of an application be physically separated.

- Allows components of each layer to be revised or enhanced without affecting the other layers.

Drawing Physical DFDs for Network Architecture

- Develop a physical data flow diagram (DFD) for the network architecture.
 - Each process symbol represents a server or class of clients.
- For each processor, develop a physical DFD to show the event processes (from Chapter 9) that are assigned to that processor.
- All but simple processes should be factored into design units and modeled as a more detailed physical DFDs.

Design Units

Design unit – a self-contained collection of processes, data stores, and data flows that share similar design characteristics.

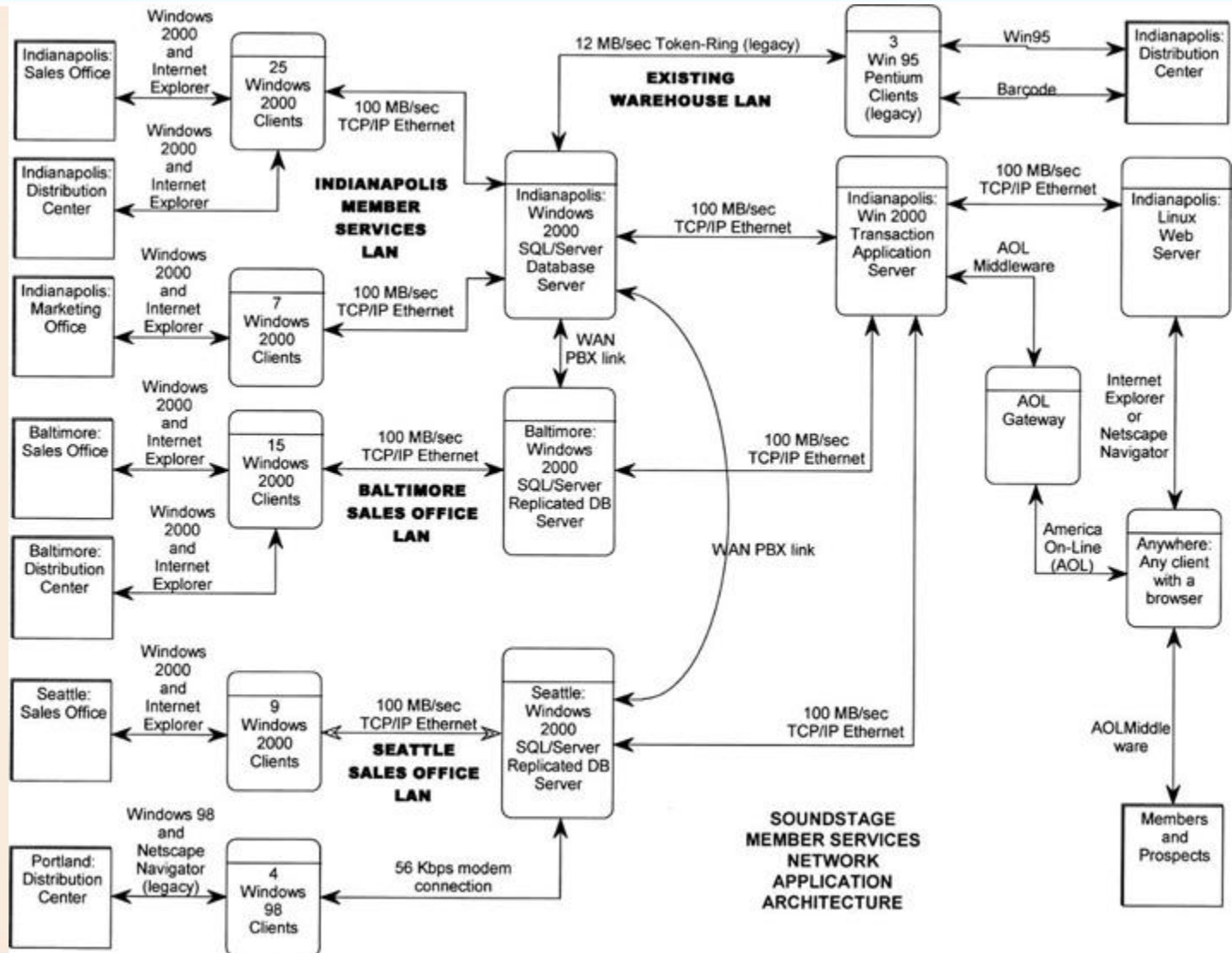
- A design unit serves as a subset of the total system whose inputs, outputs, files and databases, and programs can be designed, constructed, and tested as a self-contained unit.
- Ultimately, design units must be integrated into a whole system.

The Network Architecture DFD

Network architecture – a physical DFD that allocates processors (clients and servers) and devices (machines and robots) to a network and establishes:

- the connectivity between clients and servers
- where users will interface with the processors

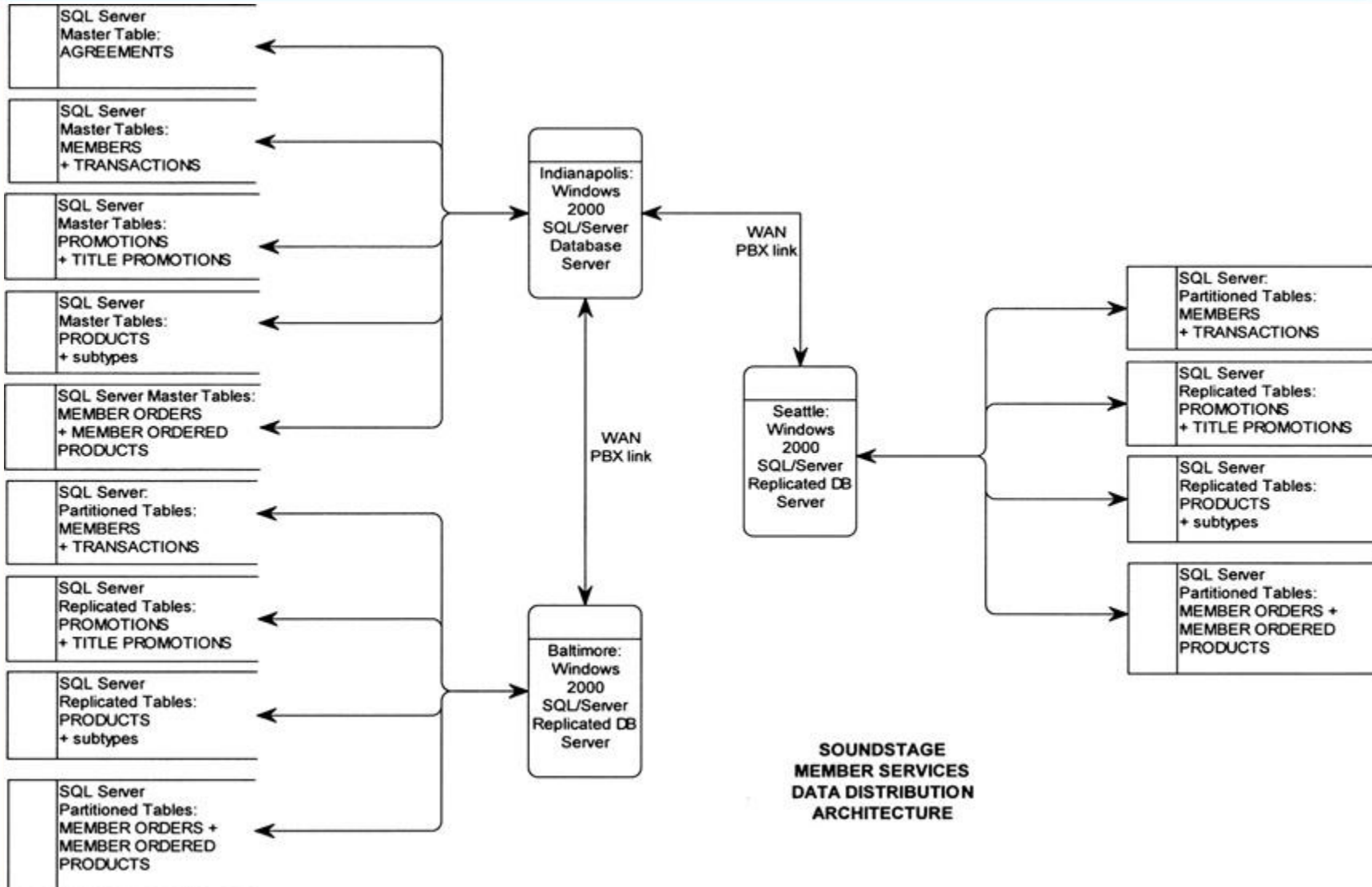
Network Architecture DFD



Data Distribution Options

- Store all data on a single server.
- Store specific tables on different servers.
- Store subsets of specific tables on different servers.
- Replicate (duplicate) specific tables or subsets on different servers.

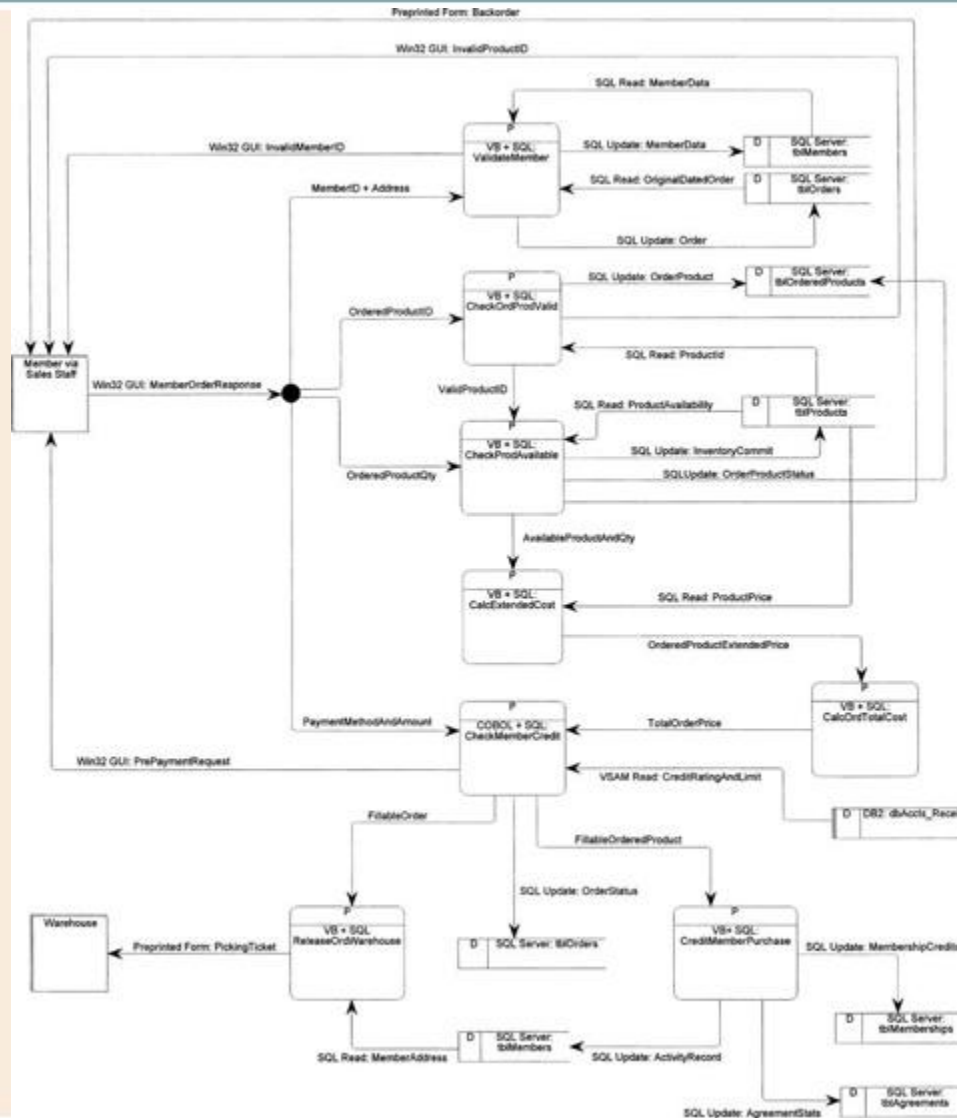
Data Distribution and Technology Assignments DFD



Process Distribution and Technology Assignments

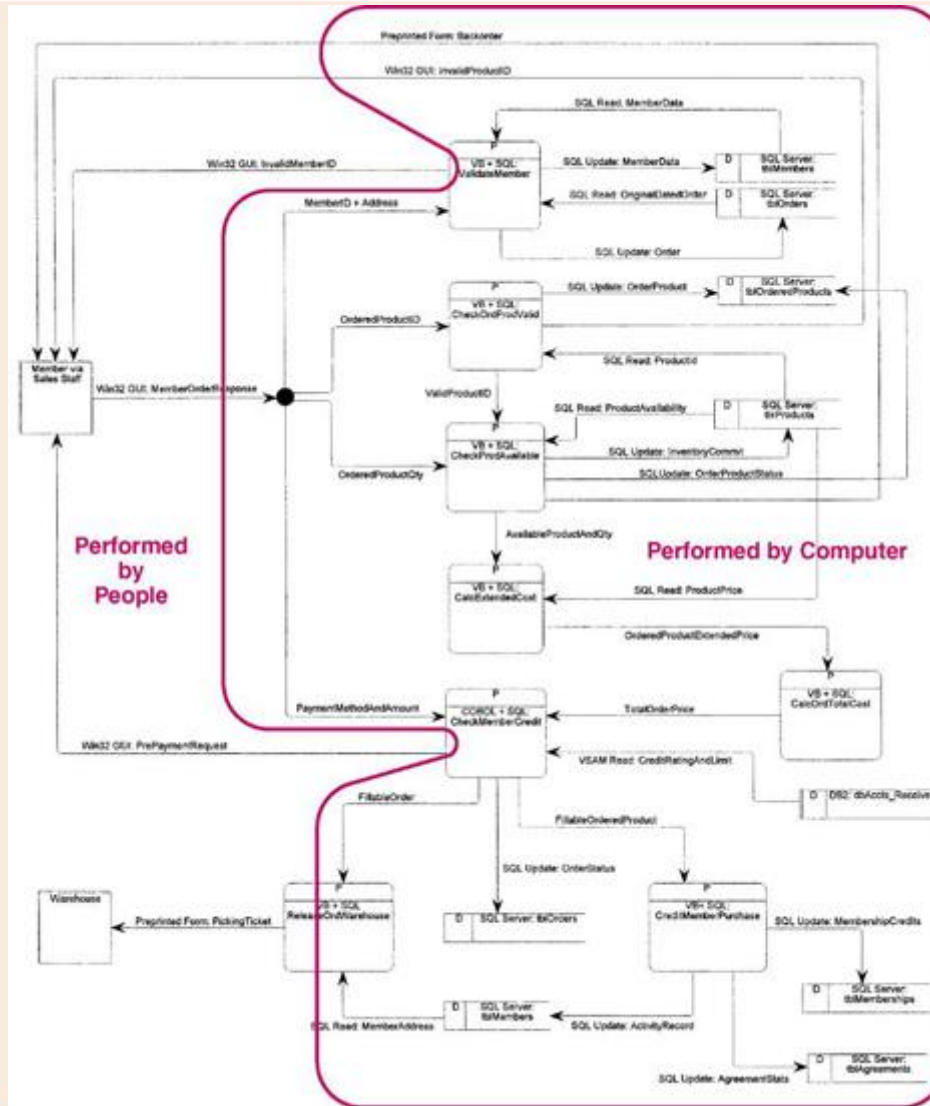
- For two-tiered client/server systems, all logical even diagrams are assigned to the client.
- For three-tiered client/server and network computing systems, must closely examine each event's primitive (detailed) DFD.
 - Determine which primitive processes should be assigned to the client and which should be assigned to an application server.
 - Generally data capture and editing are assigned to servers
 - If different aspects of a single DFD are partitioned to different clients and servers, draw separate physical DFD for each.

Physical DFD for an Event



See Figure 13-13 in text for a more readable version

The Person/Machine Boundary



See Figure 13-14 in text for a more readable version

A Manual Design Unit

