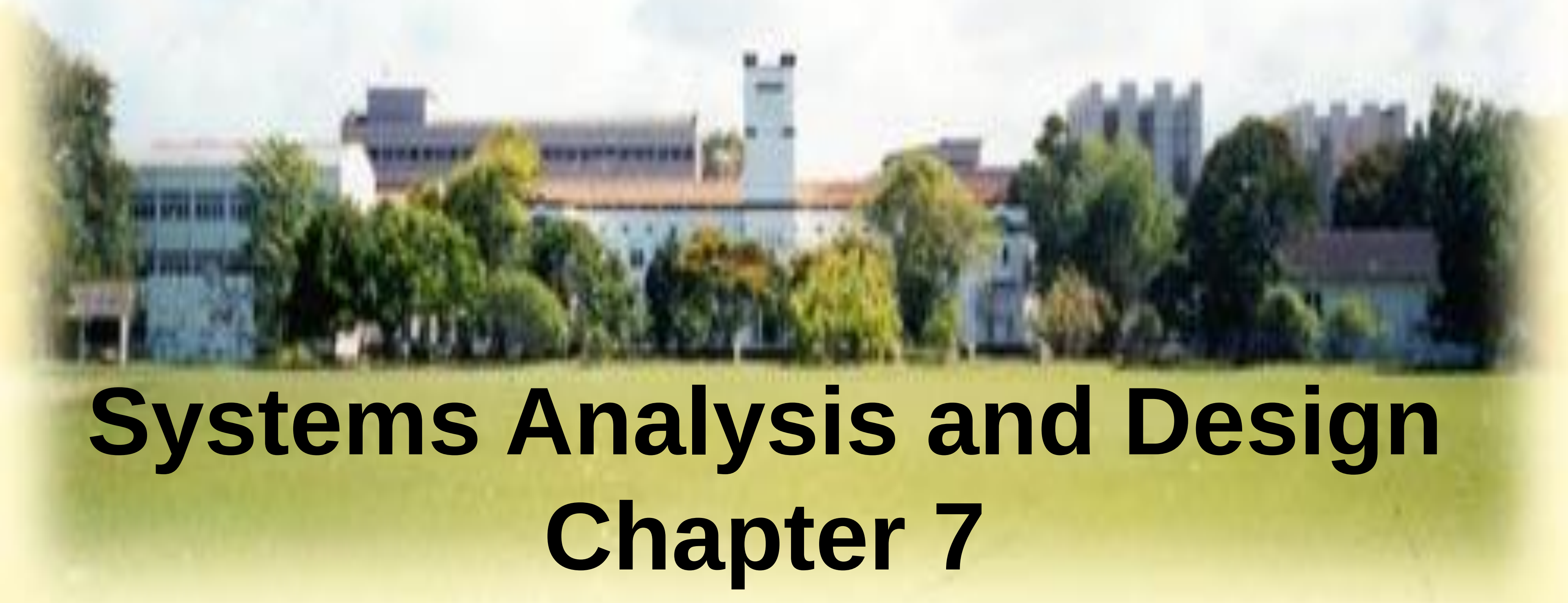
BIT 1st Year

Semester 2

IT 2405

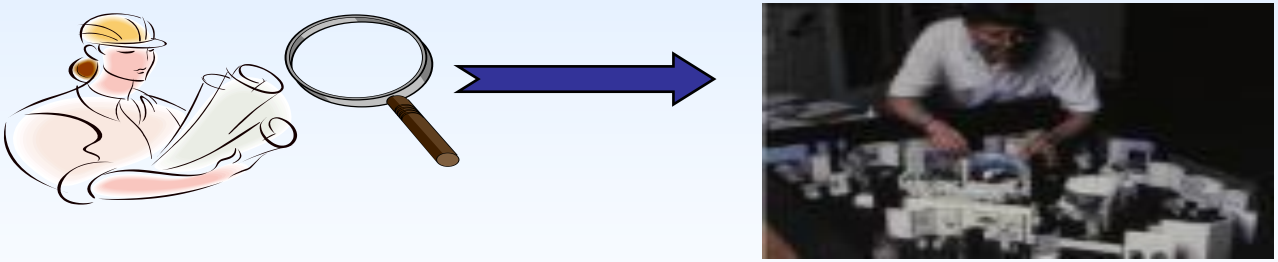


Systems Analysis and Design

Chapter 7

Systems Design

- Produces a design specification for the new system.
- Also known as physical design.
- Design inputs, outputs, files, databases and other computer based components



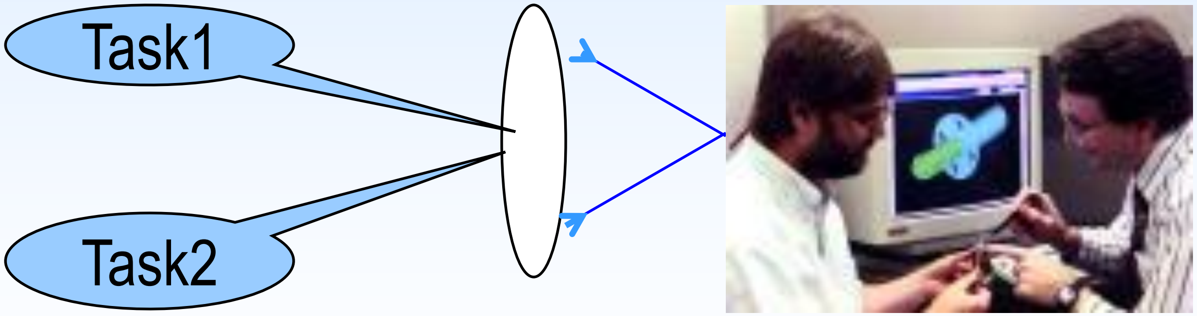
Analysis

Design

Systems Design…

Systems analysis - emphasize on the business problem

Systems design - emphasize on the technical or implementation concerns of the system.

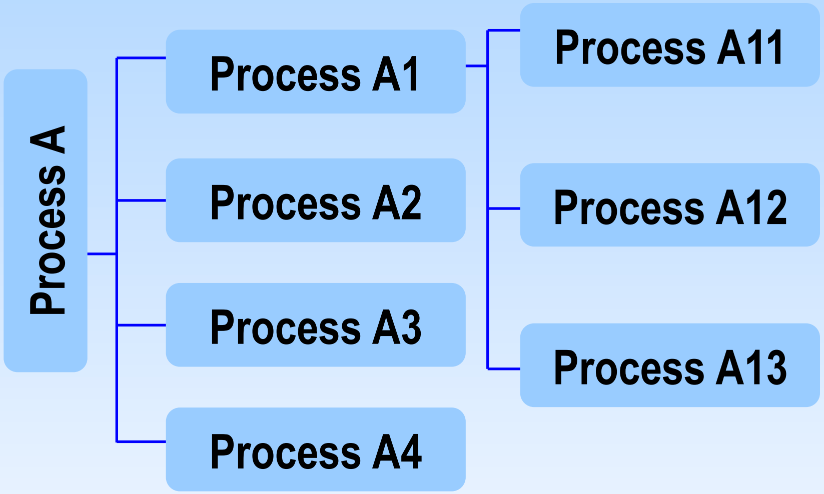


Systems Design Approaches

- Modern structured design
- Information engineering
- Prototyping
- JAD
- RAD
- Object-oriented design (Discussed in detail in Semester 3)

Modern Structured Design

- is a process-oriented technique for breaking up a large program into a hierarchy of modules



Process A

Process A1

Process A2

Process A3

Process A4

Process A11

Process A12

Process A13

- result-in a computer program that is easier to implement and maintain (change).

- synonyms are top-down program design and structured program Design.

Modern Structured Design…

- A system design technique that decomposes the system's processes into manageable components / modules that have the following properties
  - Modules should be highly cohesive (each module should accomplish one and only one function)
  - Modules should be loosely coupled (modules should be minimally dependent on one another)
  - Modules should be adaptable (It should be easy to incorporate changes)
  - Modules should be understandable

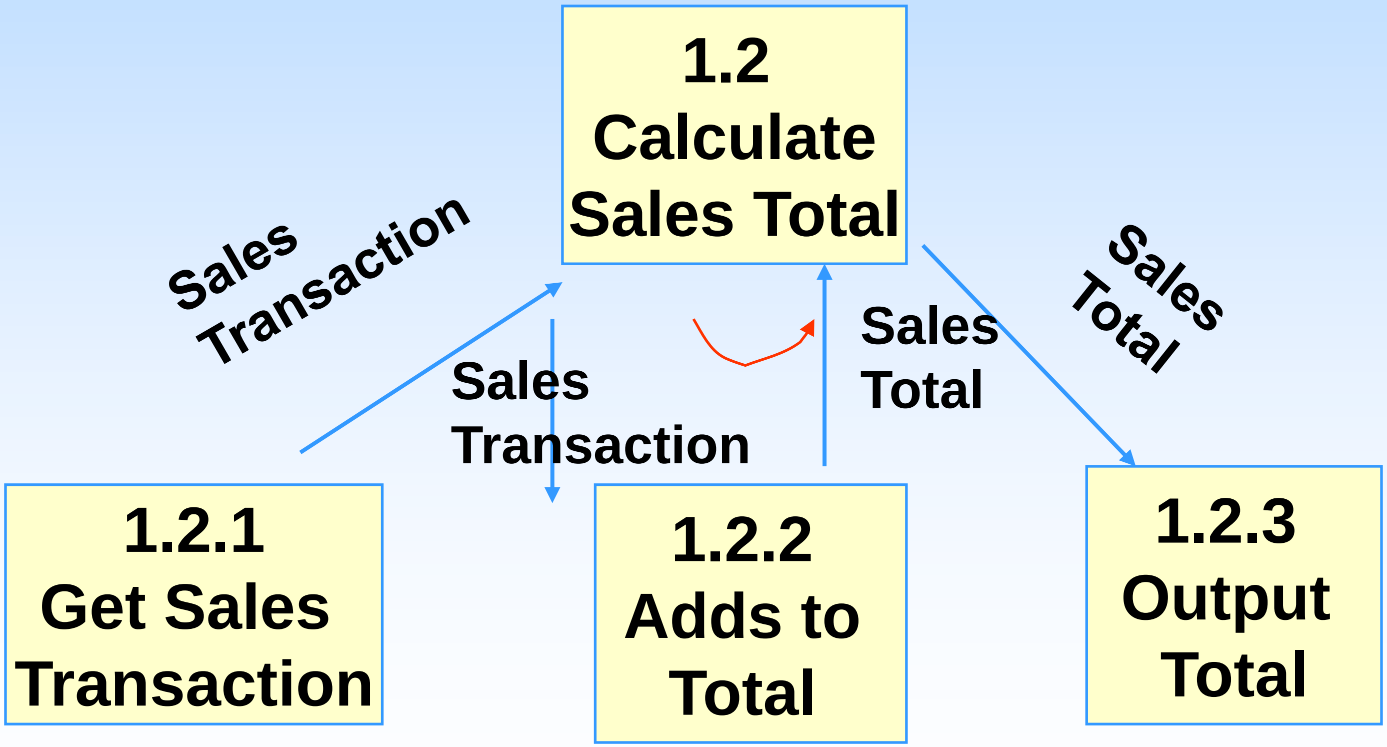
Modern Structured Design…

Structure Chart

- The software model derived from structured design
- It is derived by studying the flow of data through the program.

Modern Structured Design…

Structure Chart



Modern Structured Design…

Structure Chart

- **Parameter Passing**

-The calling module passes a set of values to the called module and receives a set of values in return. These values are passed as parameter values

eg. A value of '*sales transaction'* is passed from module *Get Sales transaction* to module *Calculate Sales Total*

Module *Calculate Sales Total* then passes the value of '*sales transaction*' to module *Add to Total* and get a value of '*sales total'* in return

The value of '*sales total'* is then passed from module *Calculate Sales Total* to module *output total*

Modern Structured Design…

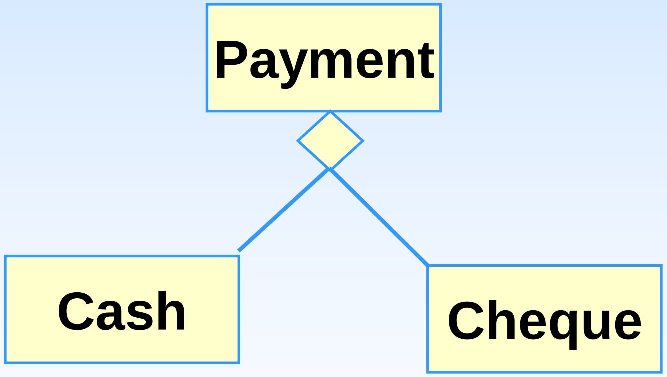
Structure Chart

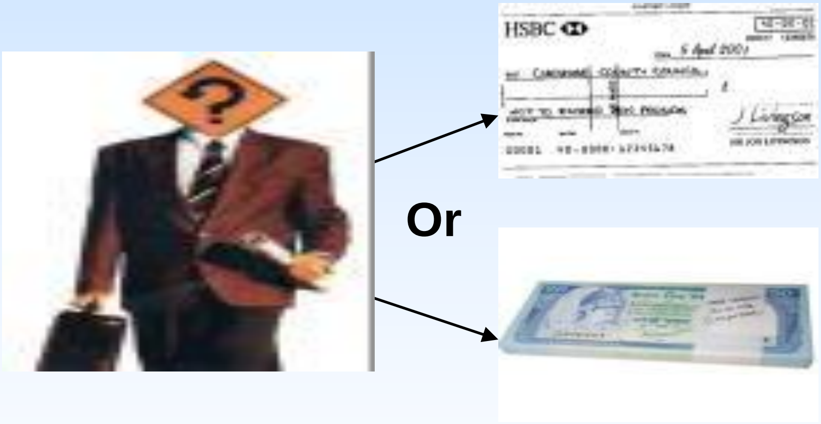
- ***Execution Sequence***
  - **By convention, modules are executed from left to right in each level.**
  - **Thus in the given example, module *Get Sales Transaction* is called before module *Adds-To-Total*. Module *Output Total* is the last module to be called.**
  - **Certain conventions are also used to represent decisions and repetition.**
  - **Decisions occur whenever a calling module has to decide to call only one of a number of modules.**

Modern Structured Design…

Structure Chart

- Decisions are modeled by a diamond symbol.





@ Payment pays either cash or Cheque

Modern Structured Design…

Structure Chart

- Repetition occurs when some modules are called repetitively by the calling module.

- Repetition is modeled by a looping arrow

Modern Structured Design…

Structure Chart is a technique used in ***Modern Structured Design***

The objective of structured design is to produce a good design.

Information Engineering (IE)

- Model driven and data centered, but PROCESS-sensitive technique for planning, analyzing, and designing information systems.
- Primary tool of information engineering is a data model diagram (ERD).
- Involves conducting a business area requirements analysis from which information system applications are carved out and prioritized.

Prototyping

Prototyping:

The prototyping approach is an iterative process involving a close working relationship between the designer and the users.

Prototyping…

Key Benefits:

- Prototyping encourages and requires active end-user participation.
- Iteration and change are a natural consequence of systems development - that is end-users tend to change their minds.
- Prototyping endorses the philosophy that end-users will not know what they want until they see it.

Prototyping…

Key Benefits:

- Prototypes are an active, model that end-users can see, touch, feel, and experience.
- An approved prototype is a working equivalent to a paper design specification, with one exception -- errors can be detected much earlier.
- Prototyping can increase creativity because it allows for quicker user feedback, which can lead to better solutions.
- Prototyping accelerates several phases of the life cycle, possibly bypassing the programmer.

Prototyping…

Disadvantages:

- Prototyping encourages ill-advised shortcuts through the life cycle.

Joint Application Development (JAD)

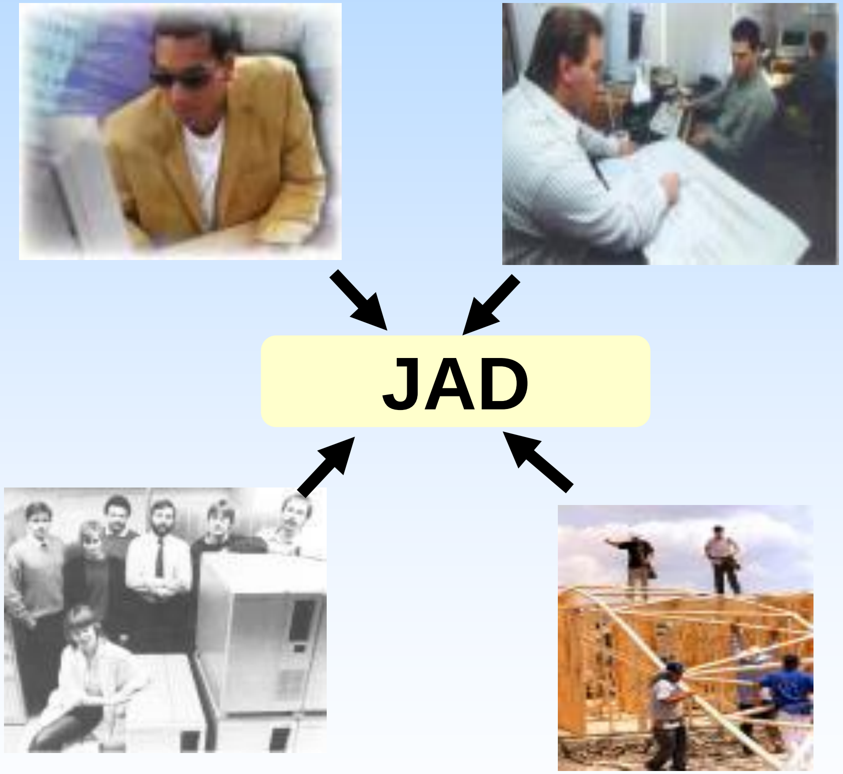
JAD emphasize *participative development* among

system owners,

users,

designers, and

builders.



JAD

Joint Application Development (JAD)…

JAD sessions for systems design,

systems designer - role of facilitator for possibly several full-day workshops intended to address different design issues and deliverables.

Rapid Application Development (RAD)

RAD is the merger of various structured techniques (especially the data-driven information engineering) with *prototyping* techniques *and joint application development* techniques to accelerate systems development.

Rapid Application Development (RAD)…

RAD calls for the interactive use of structured and prototyping to define the users' requirements and design the final system.

- Using structured techniques, the developer first builds the preliminary data and process models of the business requirements.
- Prototypes then help the analyst and users to verify those requirements and to formally refine the data and process models.

Object Oriented Design (OOD)

- The newest design strategy
- Used to refine the object requirements definitions identified earlier during analysis and to define design-specific objects
  - **e.g. based on a design implementation decision, during OOD the designer may need to revise the data or process characteristics for an object that was defined during systems analysis**

System Design Tasks

- Design the Application Architecture
- Design the System Database (s)
- Design the System Interface
- Package Design Specifications
- Update the Project Plan

Application Architecture and Modeling

Application Architecture

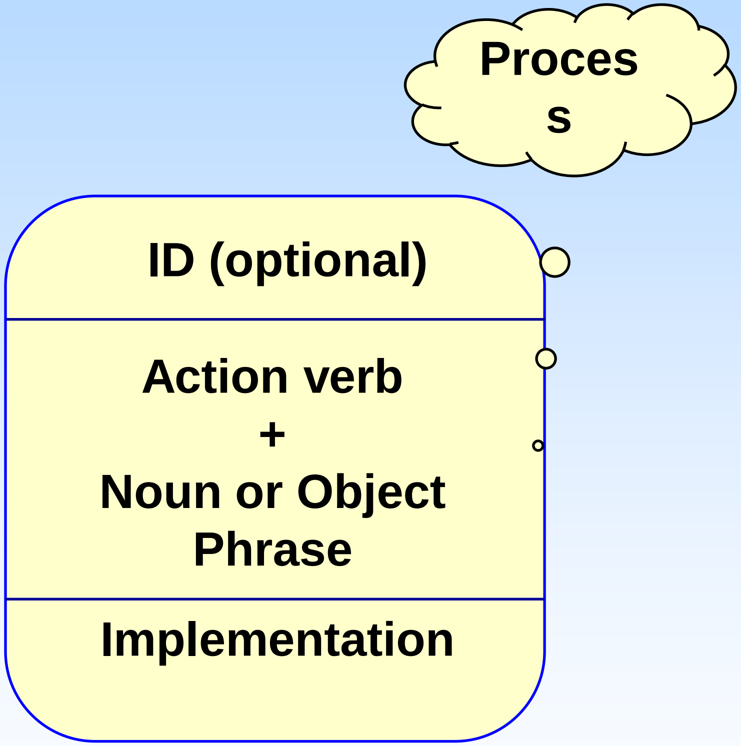
✓ **An application architecture defines the technologies to be used by (and used to build) one, more, or all information systems in terms of its *data, process, interface* and how these components interact across a network.**

✓ **It serves as an outline or blueprint for detailed design and implementation.**

✓**Primary tool : Physical data flow diagram**

Physical Data Flow Diagrams (PDFD)

- Model the technical and human decisions to be implemented as part of an information system.
- They communicate technical choices and other design decisions to those who will actually construct and implement the system.



Proces
s

ID (optional)

Action verb
+
Noun or Object
Phrase

Implementation

Physical Data Flow Diagrams (PDFD)…

Physical Process

A **physical process** is 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.

Physical Data Flow Diagrams (PDFD)…

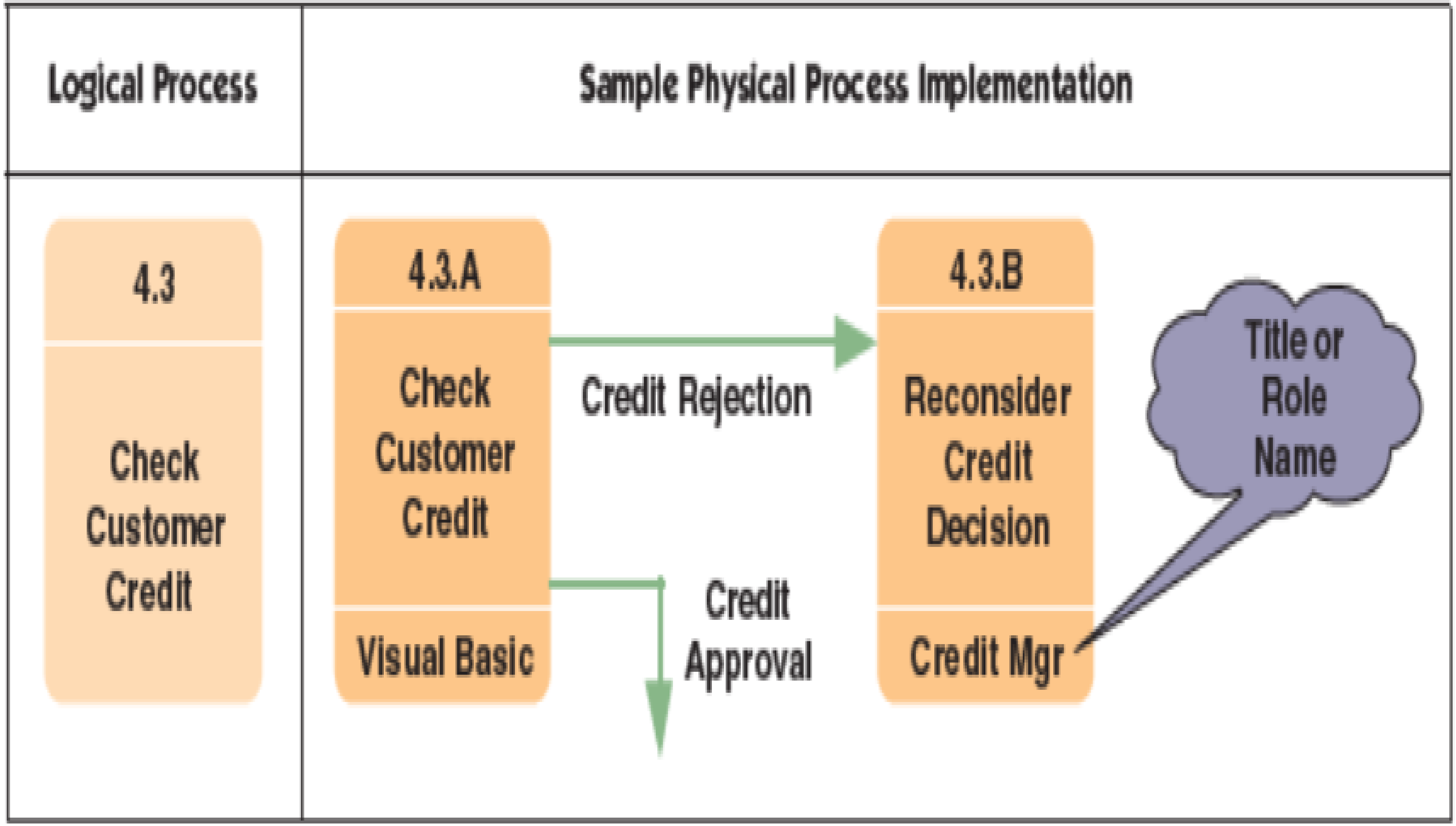
Characteristics of Physical DFDs

# Logical processes may be assigned to physical processors such as PCs, servers, mainframes, people, or devices in a network. A physical DFD would model that network structure.

# Each logical process must be implemented as one or more physical processes as some logical processes must be split into multiple physical processes.

Some logical processes may be split into multiple physical processes.

- To define those aspects that are performed by people or computers.
- # To define those aspects to be implemented by different technologies.
- # To show multiple implementations of the same process. Eg. One process for paper orders, one for internet orders
- # To add processes for exceptions and internal control (e.g., security).



Logical Process

Sample Physical Process Implementation

4.3

Check Customer Credit

4.3.A

Check Customer Credit

Visual Basic

Credit Rejection

Credit Approval

4.3.B

Reconsider Credit Decision

Credit Mgr

Title or Role Name

Physical Data Flow Diagrams (PDFD)…

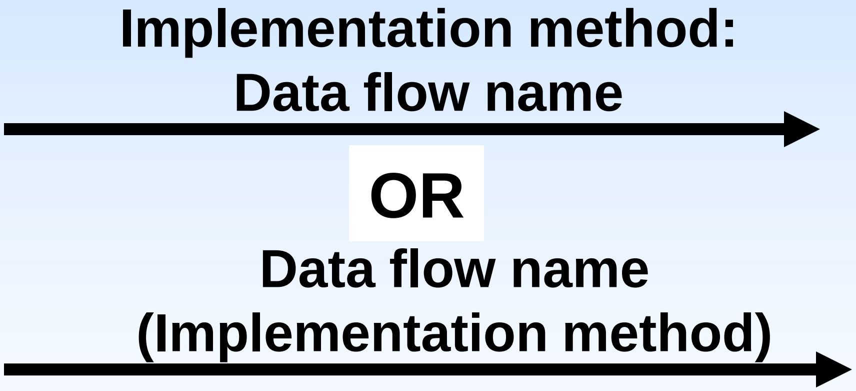
Physical Data Flows

- Represents any of the following:
  - Planned implementation of an input to / output from a physical process.
  - A database command or action (create, read, update, or delete)
  - Import of data from or the export of data to another information system across a network.
  - Flow of data between two modules within the same program.

Physical Data Flow Diagrams (PDFD)…

Physical Data Flows…

Physical Data Flow Representation



Implementation method:
Data flow name

OR

Data flow name
(Implementation method)

Refer Figure 13-2 pg 482 Ref1 to learn how to apply one of these naming conventions in physical DFDs

eg.

PRINTOUT:
Salary Equity Report

Physical Data Flow Diagrams (PDFD)…

Physical External Agents

- **No change from logical DFD to Physical DFD.**
  - **External agents were classified during systems analysis as outside the scope of the systems and therefore, not subject to change.**
  - **Only a change in requirements can initiate a change in external agents**

Physical Data Flow Diagrams (PDFD)…

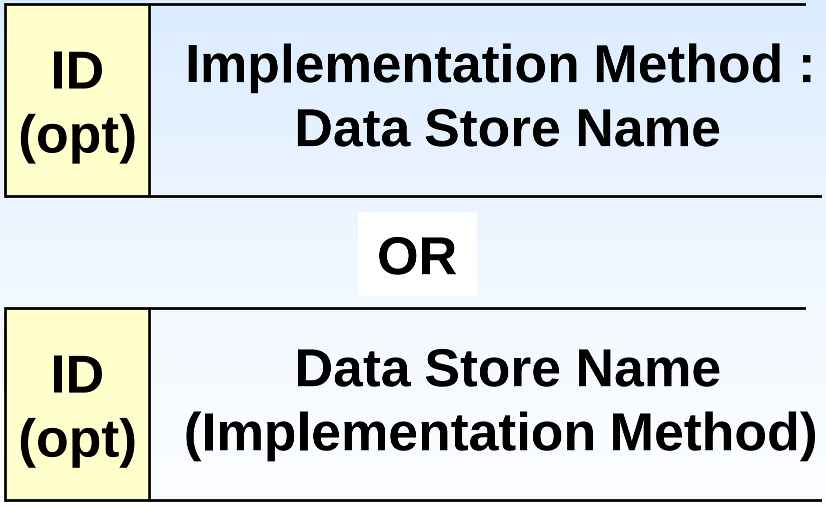
Physical Data Stores

- **Represents the implementation of one of the following:**
  - **A database**
  - **A table in a database**
  - **A file (computer/non computerized)**
  - **A tape / Media backup of anything important**
  - **Temporary file needed by a program (e.g. Tax Tables)**

Physical Data Flow Diagrams (PDFD)…

Physical Data Stores

Representation of physical data stores



ID (opt)

Implementation Method : Data Store Name

OR

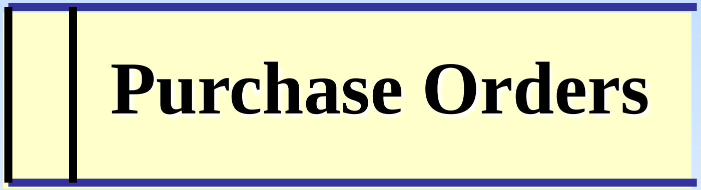
ID (opt)

Data Store Name (Implementation Method)

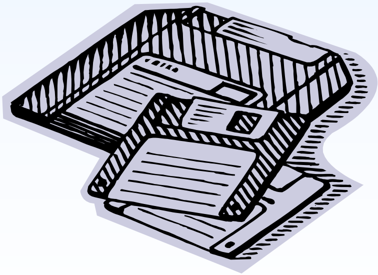
Physical Data Flow Diagrams (PDFD)…

Physical Data Stores

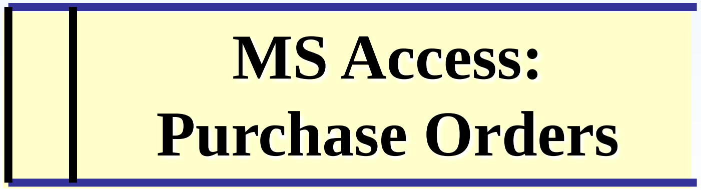
Logical Data store


Purchase Orders

Implementation : A table in a database



Physical Data store


MS Access:
Purchase Orders

Design the System Database

- Databases are a shared resource.
- A database should be adaptable to future requirements and expansion.
- Issues to be addressed during database design include
  - how programs will access the data
  - Programming data structures and their impact on performance and flexibility
  - Internal controls to ensure proper security and disaster recovery technique, in case data is lost or destroyed.
  - Record size and storage volume requirement.

Design the System Database…

- Purpose is to prepare technical design specification for the database.
- Participants
  - – **Systems analyst – participate in database modeling**
  - – **System designers – complete the database design**
  - – **Data administrator – recognize that the new system most likely use s some portion of an existing database**
  - – **System builders – build a prototype database for the project**
- Inputs : **The application architecture and Distributed analysis decisions**
- Output / deliverable : **Database schema**

Design the System Interface

- Designer can work closely with system user to develop input, output and dialogue specifications.
- The precise format and layout of the outputs must be specified.
- Internal controllers must be specified to ensure that the outputs are not lost, misrouted, misused, or incomplete.
- the data capture methods for the inputs should be designed.
- Editing controllers must be designed to ensure the accuracy of input data.

Design the System Interface…

- For dialogue design the designer must consider
  - **Terminal familiarity**
  - **Possible errors and misunderstandings that the end user may have or may encounter**
  - **The need for additional instructions or help at certain points**
  - **The screen content and layout**
- Participants
  - **System users**
  - **System designers**
  - **System builders**

Package Design Specifications

- Package all the specifications from the previous design tasks into a set of specifications.
- Guide the computer programmers activities.
- Inputs : database, input, and output specifications

Update the Project Plan

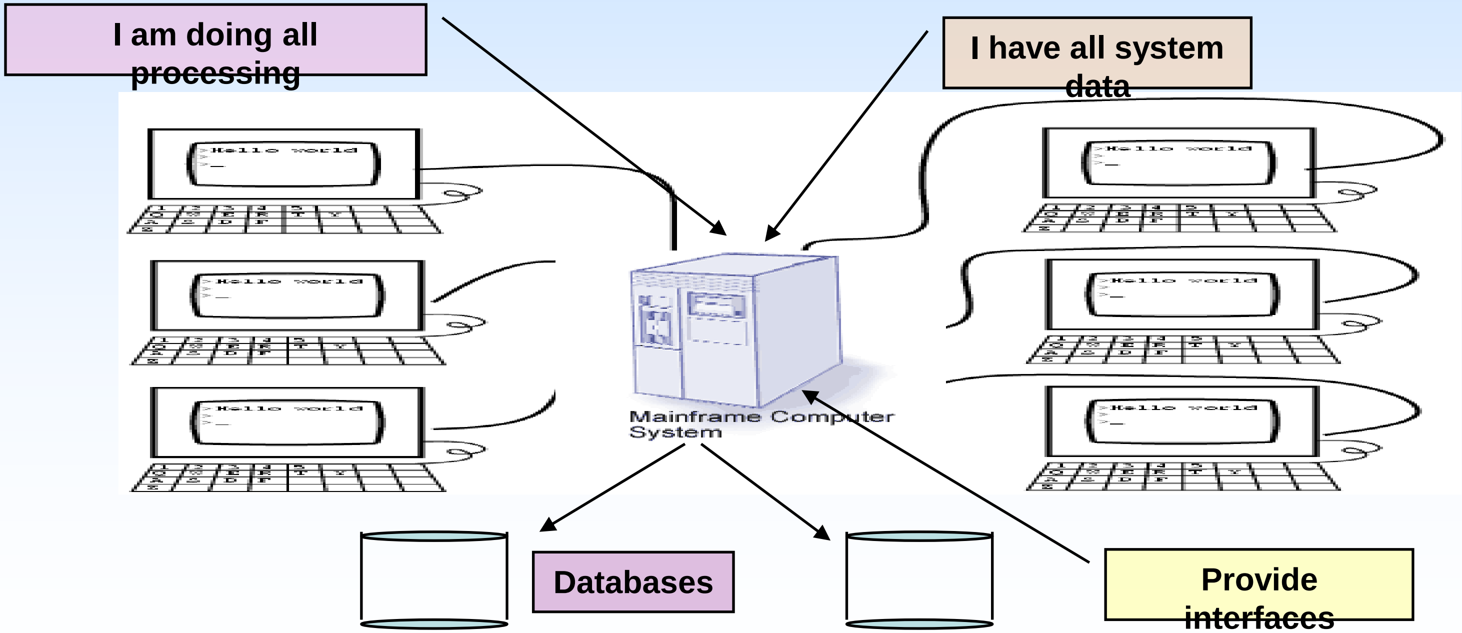
- Reevaluate project feasibility & update project plan
- Project manager facilitates the task
- Analysts and owners should consider the possibility that, based on the completed design work, the overall project schedule, cost estimates, and other estimates may need to be adjusted.
- The deliverable : updated project plan
  - **Should include a detailed plan for the construction phase that should follow.**

Information Technology Architecture

- System analysts must continuously read popular trade journals to stay abreast of the latest technologies and techniques that will keep their customers and their information systems competitive.
- The information system framework provides one suitable framework for understanding IT architecture.
- Today information systems are
  - no longer monolithic, mainframe computer based systems.
  - Built on some combination of networks (Distributed)

Centralized Systems

- All the components are hosted by a central, multi user system
- User interact with the host computer via terminals
- All of the actual processing and work is done on the host computer



Distributed Systems

- Components are distributed across multiple locations and computer networks
- Processing work load required to support these components are distributed across multiple computers on the net work.
- More complicated and more difficult to implement than centralized systems

Distributed Systems…

- Why use distributed systems?
  - Modern businesses are already distributed
  - Distributed computing moves information and services closer to the customers
  - Consolidates the incredible power
  - More user friendly as they use the PC as the user interface processor
  - PCs and network servers are much less expensive than mainframes

Thus, there is a big trend towards distributed systems.

Distributed Systems…

- Disadvantages
  - Network data traffic can cause congestion that actually slows performance.
  - Higher security risk due to more possible access points for intruders and possible communication with insecure systems.

Many centralized, legacy systems are gradually being transformed into distributed information systems

Distributed Systems…

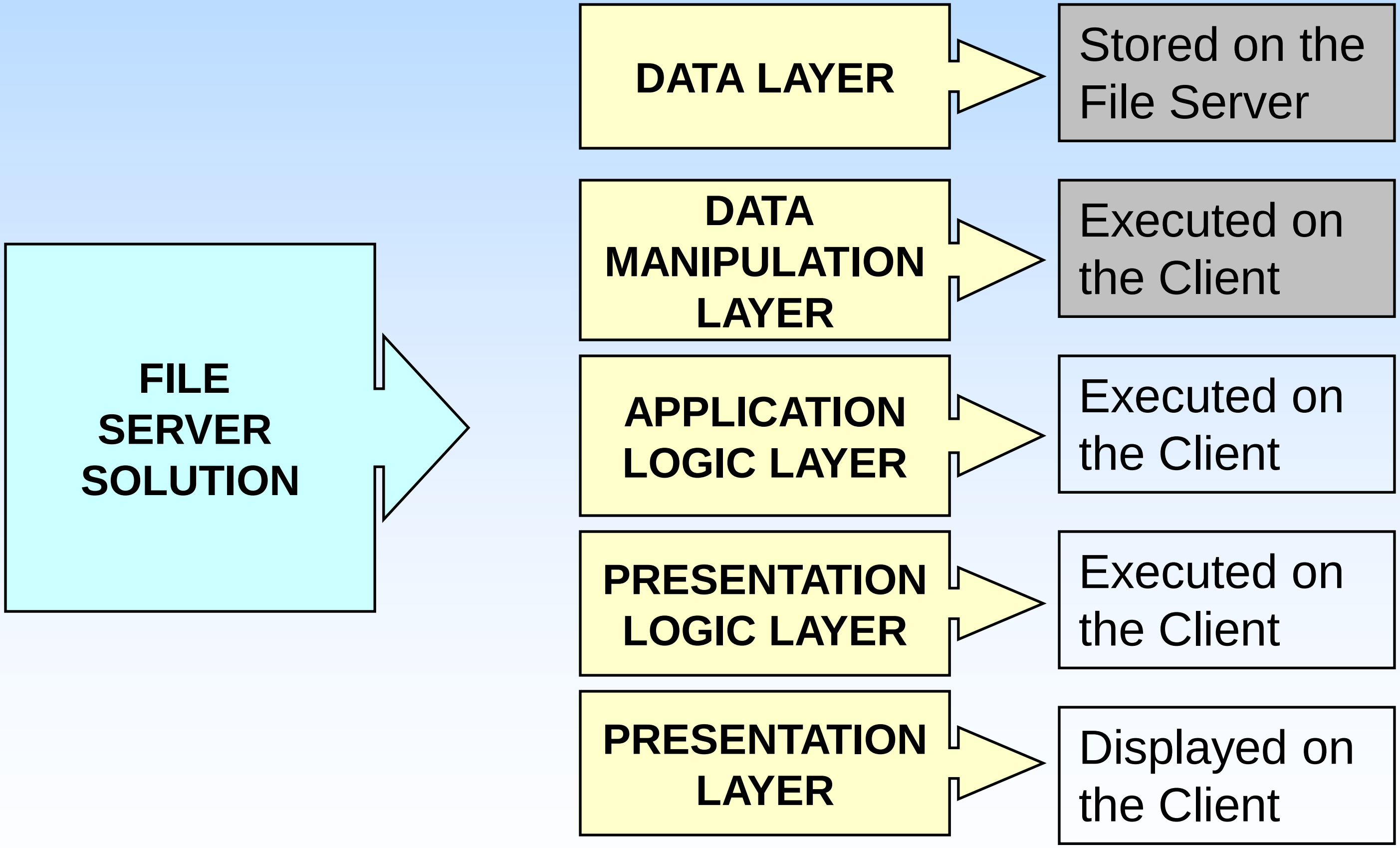
Architecture Layers

- Presentation layer – actual user interface which presents inputs and outputs to the user
- Presentation logic layer – any processing that must be done to generate the presentation. e.g. editing input data, formatting output data
- Application logic layer– all the logic and processing required to support the actual business application and rules. e.g. credit checking, calculations, data analysis
- Data manipulation layer – commands and logic required to store and retrieve data to and from the database
- Data layer – the actual stored data in the in a database

Distributed Systems…

- There are three types of distributed system architectures
  - File server architecture
  - Client server architecture
  - Internet based architecture

Distributed Systems…



FILE SERVER SOLUTION

DATA LAYER	Stored on the File Server
DATA MANIPULATION LAYER	Executed on the Client
APPLICATION LOGIC LAYER	Executed on the Client
PRESENTATION LOGIC LAYER	Executed on the Client
PRESENTATION LAYER	Displayed on the Client

Distributed Systems…

File server Architecture

- A LAN (Local area network) based solution
  - LAN is a set of client computers connected over a relatively short distance to one or more servers
- A server computer hosts only the data layer
- All other layers are implemented on the client PC.
- Practical only for small database applications shared by relatively few users

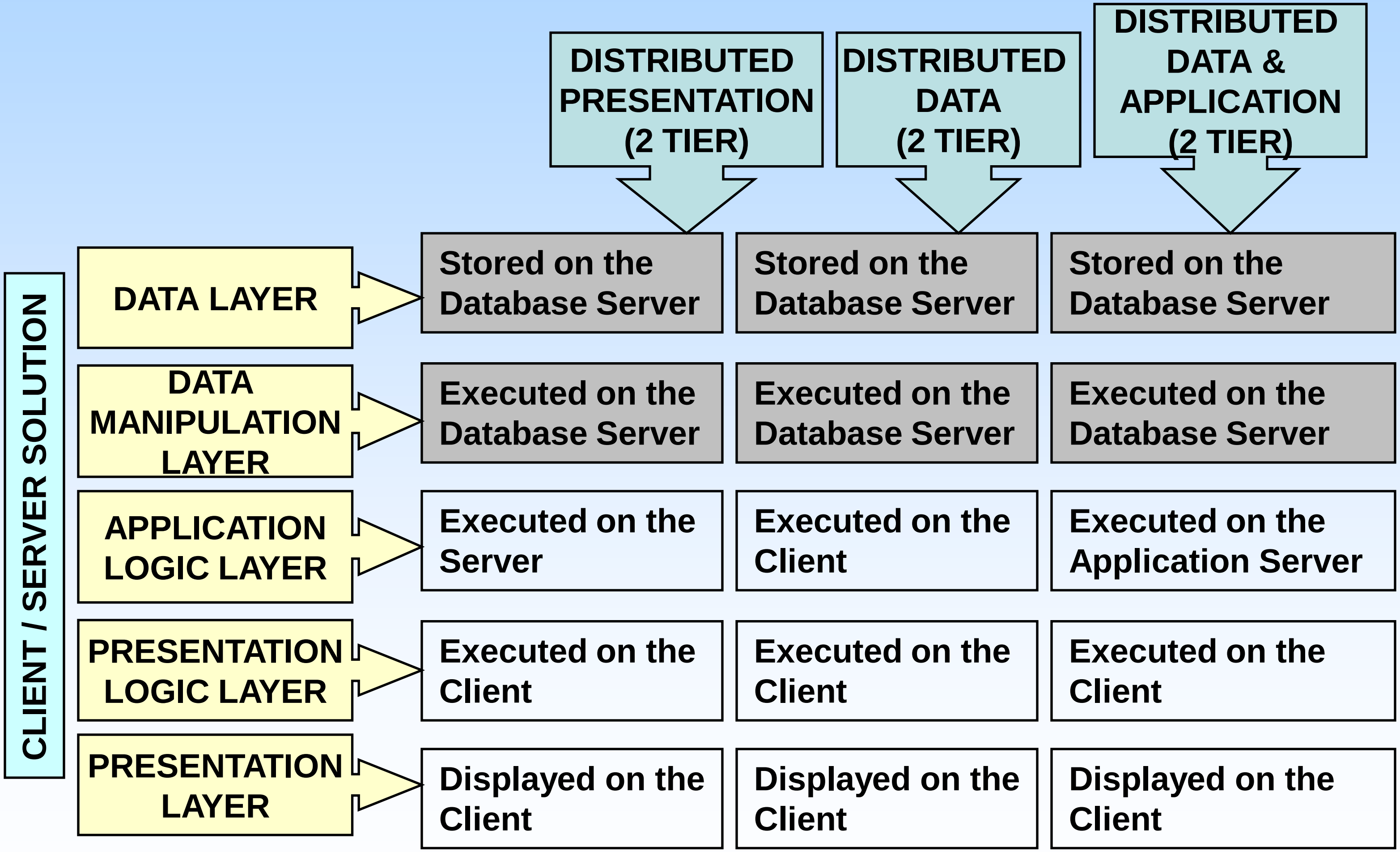
Distributed Systems…

File server Architecture…

Disadvantages

- Large amount of unnecessary data must be moved between the client and the server
- Reduce network and application performance
- The client PC must be robust.
- Data base integrity can be easily compromised

Distributed Systems…



CLIENT / SERVER SOLUTION

	DISTRIBUTED PRESENTATION (2 TIER)	DISTRIBUTED DATA (2 TIER)	DISTRIBUTED DATA & APPLICATION (2 TIER)
DATA LAYER	Stored on the Database Server	Stored on the Database Server	Stored on the Database Server
DATA MANIPULATION LAYER	Executed on the Database Server	Executed on the Database Server	Executed on the Database Server
APPLICATION LOGIC LAYER	Executed on the Server	Executed on the Client	Executed on the Application Server
PRESENTATION LOGIC LAYER	Executed on the Client	Executed on the Client	Executed on the Client
PRESENTATION LAYER	Displayed on the Client	Displayed on the Client	Displayed on the Client

Distributed Systems…

Client Server Architecture

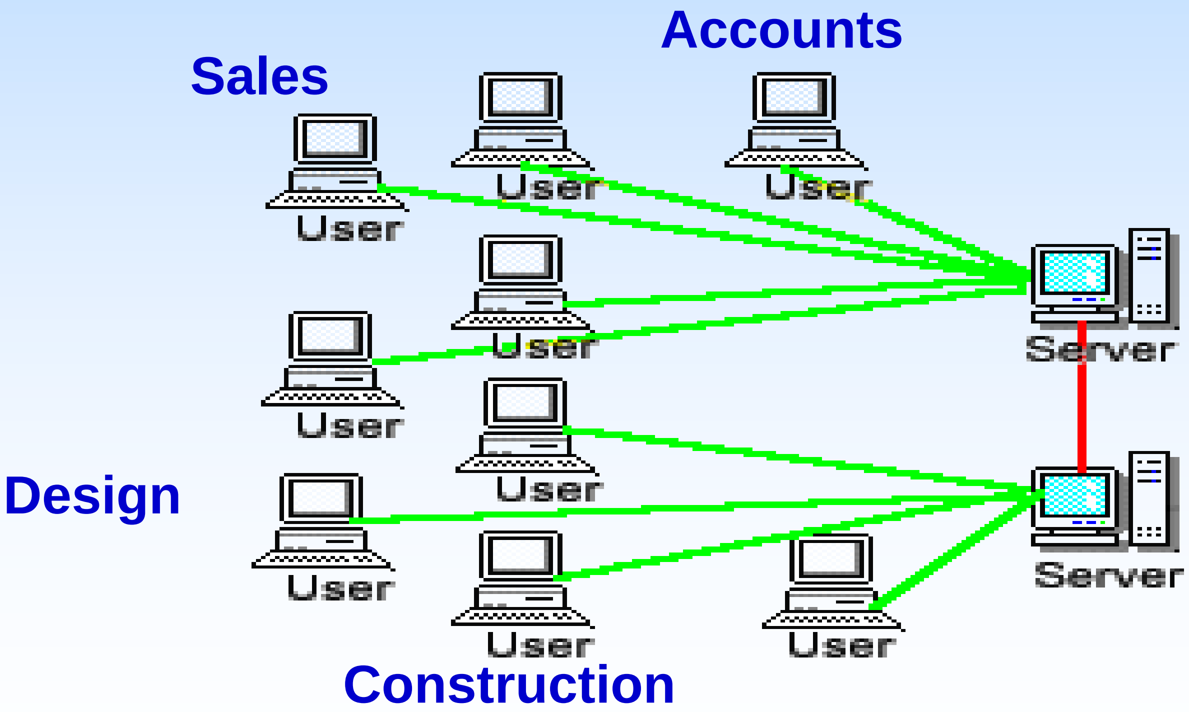
The presentation, presentation logic, application logic, data manipulation and data layers are distributed between client PC's and one or more servers.

Client Computers :

Any combination of personal Computers or Workstations, "sometimes connected"

Distributed Systems…

Client/Server Architecture…

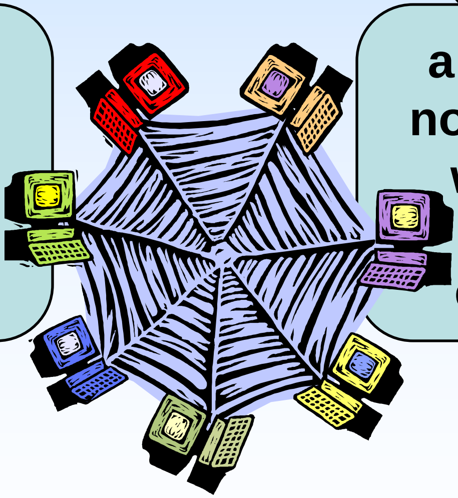


Distributed Systems…

Client/Server Architecture…

- Clients may be thin or flat

A personal that does not have to be very powerful (acts as a terminal)
e.g. Remote desktop



a personal computer, notebook computer, or work station that is typically powerful
e.g. Almost all PCs

Distributed Systems…

Client/Server Architecture…

- **Server must be more powerful and capable than a server in the file server model**
- **Several types of servers may be used in a client/server solution.**
  - **Database Server : A server that hosts one or more databases.**
  - **Transaction Server : a server that hosts services which ensure that all database updates for a transaction succeed or fail as a whole.**

Distributed Systems…

Client/Server Architecture…

- Application Server : A server that hosts application logic and services for an information system
- Messaging or Groupware Server : A server that hosts services for groupware.
- Web Server : A server that hosts internet or intranet web sites

Distributed Systems…

Client/Server Architecture…

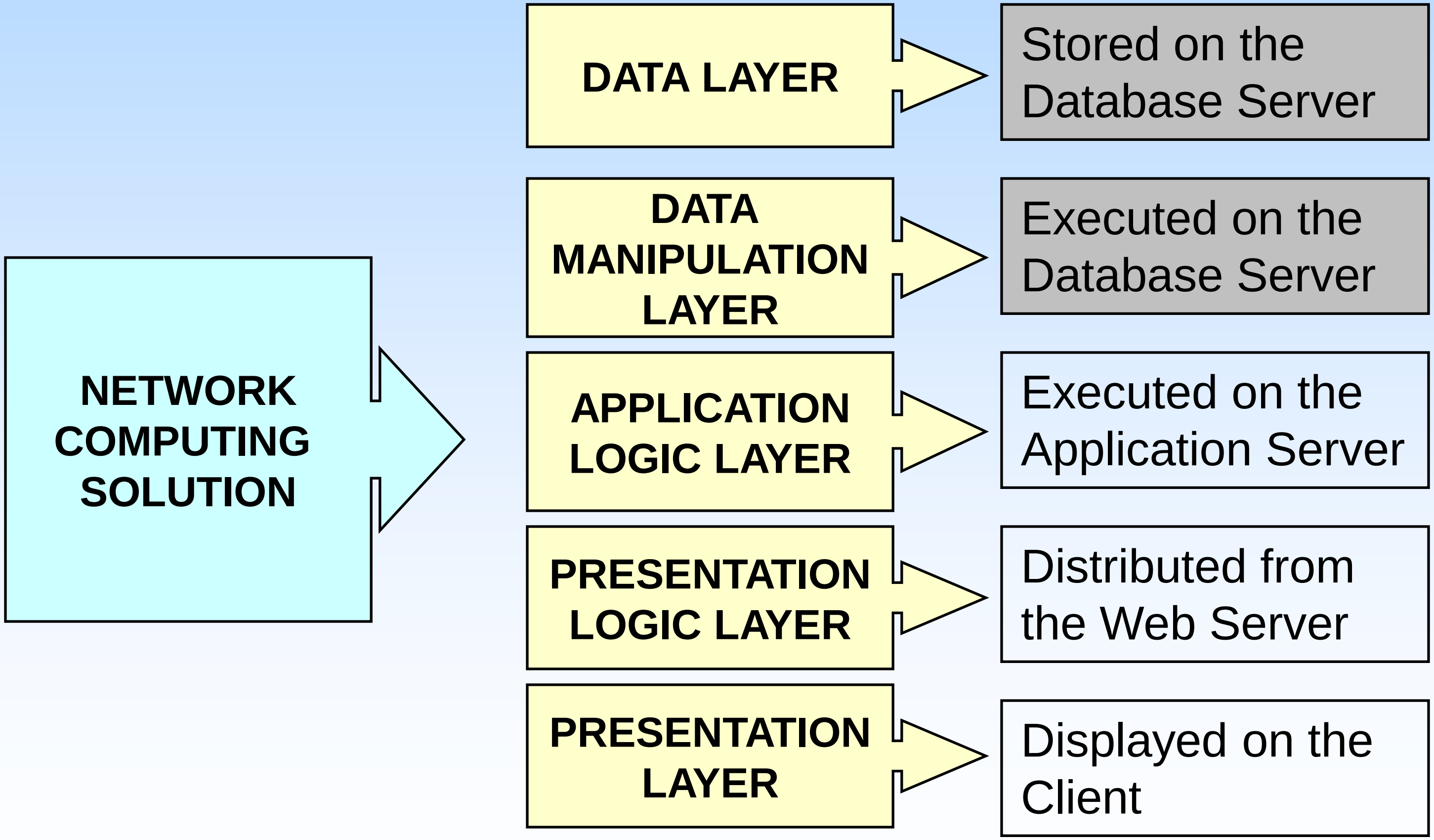
- Client / Server – Distributed Presentation
  - A client/server system in which presentation and presentation logic are shifted from the server to reside on the client
- Client / Server – Distributed Data
  - A client/server system in which the data and data manipulation layers are placed on servers and other layers are placed on clients. Also called two-tiered client/server computing.

Distributed Systems…

Client/Server Architecture…

- Client / Server – Distributed Data and Application
  - client/server system in which the data and data manipulation layers are placed on their own server (s).
  - Application logic is placed on its own server
  - Presentation logic and Presentation are placed on the clients.
  - Also, called three-tiered, or n-tiered, client/server computing

Distributed Systems…



NETWORK COMPUTING SOLUTION

DATA LAYER → Stored on the Database Server

DATA MANIPULATION LAYER → Executed on the Database Server

APPLICATION LOGIC LAYER → Executed on the Application Server

PRESENTATION LOGIC LAYER → Distributed from the Web Server

PRESENTATION LAYER → Displayed on the Client

Distributed Systems…

Internet-based Architecture

- Presentation and presentation logic layers are implemented in a client side web browser
- The presentation logic layer then connects to the application logic layer that run on an application server,
- Subsequently connects to the database server/s

Data Architectures

- Client-server and network computing made it possible to distribute data without loss of control
- Control is accomplished through advances in distributed relational database technology

Distributed Relational Database

- Stores data in a tabular form
  - Each file is implemented as a table
  - Each field is a column in the table
  - Each record is a row in the table
  - Related records between two tables (e.g. CUSTOMER and ORDER) are implemented by internally duplicating columns in the two tables.
- Distributes or duplicates tables to multiple database servers located in important locations

Distributed Relational Database Management System (RDBMS)

- The software required to implement distributed relational databases
- Controls access to and maintenance of the stored data in the relational format
- Provides back-ups, recovery and security
- Also called as client-server database management systems

e.g. Oracle, SQL Server, Sybase

Distributed Relational Database Management System (RDBMS)

- Supports two types of data
  - Data partitioning : truly distributes rows and columns to specific database servers with little or no duplication between servers
  - Data replication : duplicates some or all tables on more than one database server.

Data Architectures

- For a given information system application the data architecture must specify the RDBMS technology and the degree to which data will be partitioned or replicated.
- One way to record these decisions is to record them in a physical data store

For more information Refer pg 495 Ref1

Interface Architectures

Batch inputs or Outputs

- Transactions are accumulated into batches for periodic processing
- Batch inputs (e.g. time cards) are processed to update databases and produce appropriate outputs (e.g. paychecks, generation of invoices)

Refer pg 496 Ref1 for more information

Interface Architectures…

Online inputs or Outputs

- Provide for a more conversational dialogue between the user and the computer applications.
- Provide immediate feedback in response to transactions, problems, and inquiries.
- Provide greater human interaction in decision

Refer pg 497 Ref1 for more information

Interface Architectures…

Remote batch

- Combines the best aspects of batch and online inputs and outputs.

  Refer pg 497 Ref1 for more information

Interface Architectures…

Keyless Data Entry (and automatic identification)

- Keying in data has always been a major source of errors in computer inputs.
- In batch systems, keying errors can be eliminated through optical character reading (OCR) and optical mark reading (OMR) technology
- The real advance in keyless data entry are coming for online systems in the form of auto-identification systems. (e.g. bar-coding schemes)

Refer pg 498 Ref1 for more information

Interface Architectures…

Pen input

- A pen-based operating system become more widely available and used (e.g. Palm OS and Microsoft’s Windows Mobile)
- Uses this technology to for remote data collection

Refer pg 498-499 Ref1 for more information

Interface Architectures…

Electronic Data interchange (EDI)

- The standardized electronic flow of business transactions or data between businesses

Refer pg 499 Ref1 for more information

Interface Architectures…

Imaging and Document Interchange

- The actual images of the forms and data are transmitted and received.
- Useful in applications in which the form images or graphics are required

Refer pg 499 Ref1 for more information

Interface Architectures…

Middleware

- Utility software that enables communication between different processors in a system
- May be built into the respective operating system or added through purchased middleware products

Refer pg 500 Ref1 for more information

Process Architectures

The software development environment (SDE)

- A language and tool kit for developing applications
- One way to classify SDEs is according to the type of client/server or network computing architectures they support
  - **SDEs for Centralized Computing and Distributed Presentation**
  - **SDEs for Two-Tier Client/Server**
  - **SDEs for Multi-tier Client/Server**
  - **SDEs for Internet and Intranet Client/Server**

Refer pg 500-502 Ref1 for more information