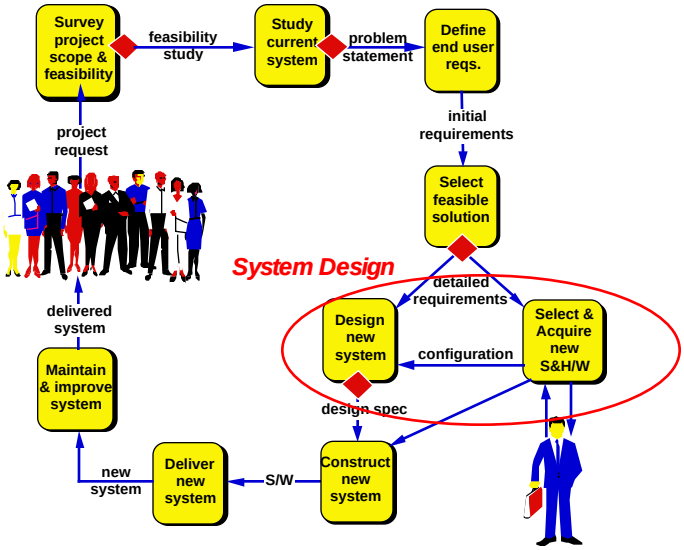
XVI. System Design

What is System Design?
The Outputs of System Design
The Global System Architecture
Classification of Applications
State of the Market

Back to the Information System Lifecycle



Major Concerns of System Design

- Identify major subsystems and components.
- Identify (usage, control or data) dependencies among subsystems.
- Decide on a hardware and software platform for the new system, I.e., the hardware and network(s) on which it will run, the operating system and other off-the-shelf software (e.g., DBMSs) it will use.
- Allocate subsystems to hardware nodes (for a possibly distributed system.)
- Decide on a data management strategy.
- Choose a strategy and standards for human-computer interfaces.

Other Elements of System Design

(...Not discussed in this course….)

- Plan control aspects of the application.
- Produce test plans.
- Specify code development standards.
- Set priorities for design trade-offs.
- Identify implementation requirements (e.g., data conversion)

Outputs of the Design Phase

- Hardware, networking and software platform for the new system.
- A (global) system architecture, which describes the hardware nodes and communication connections among them.
- A software architecture for the new system, showing the hierarchy of subsystems and their inter-dependencies.
- An allocation of subsystems and data to hardware nodes.
- A detailed description of interactions between different elements of the design (through sequence, collaboration, state and activity diagrams.)
- A database design, consisting of a database schema for the data managed by the new system.
- User interfaces for different groups of users.

Global System Architecture

- Describes the collection of inter-connected hardware nodes on which the system will eventually run.
- A global system consists of:
- Hardware nodes, where components of the new system will run; for each node select a hardware configuration and operating system platform that will run; for example, hardware platform: 486, 2MB RAM, 100MB disk OS: DOS Windows.
- The connectivity among hardware nodes, defined by length of connection, type of connection, product used for the connection; for example,

  length: <100ft, 100miles

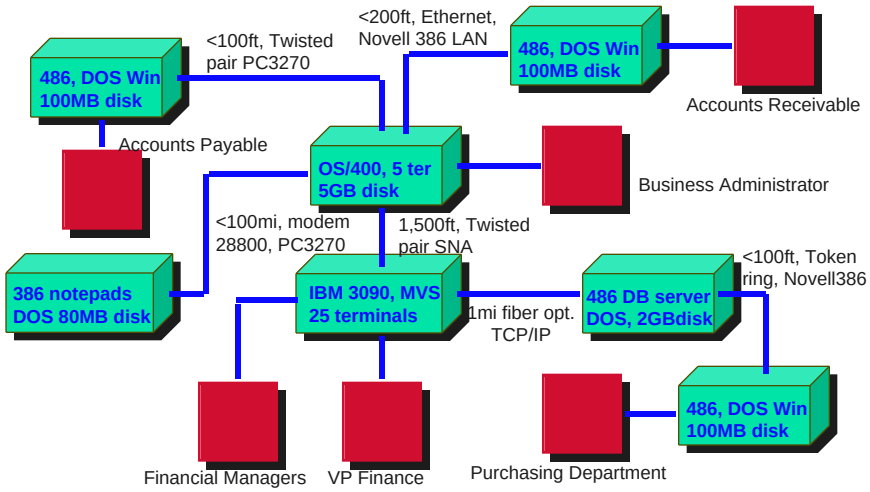
  type: twisted pair, fiber optic, ethernet

  product: Novell 386 LAN, PC3270
- The location of users, inputs and outputs for the new system;

Key concern: Minimize data communication

Example Global Distributed Architecture

- To each hardware node, associate users (external entities) and network interconnections



Distribution Issues: Inputs and Outputs

- Batch mode -- process a batch of inputs/outputs together; sometimes most appropriate solution
  e.g., incoming mail (purchase orders), outgoing mail (invoices, cheques)
- On-line mode -- process inputs/outputs as they become available; can save data entry time, particularly if end user can do the input, clearly the way of the future, because on-line data entry can be done on PCs
- Remote batch -- data are input on-line on several machines, then fed in a batch mode to a centralized database

New Technologies for I/O and New Standards for Data Interchange

- Keyless data entry -- bar coding, optical character recognition, special keyboards -- very appropriate for large volumes of I/O
- Pen input -- several products in the market, with mixed success rate for different types of data
- Electronic data interchange (EDI) -- data are transferred through telephone lines from one location to another
  e.g., credit card charging
- Image and Document Interchange -- like electronic data interchange, but now whole documents, including images, are passed around
  e.g., law enforcement, bank applications
- HTML/XML/SGML -- markup languages for documents; SGML is a general markup languages for documents; HTML is a special version used for WWW documents; XML is something in between.

Deciding on a Global System Architecture

Here is a series of issues that need to be addressed:

- Establish batch and on-line computer processes; key consideration: data communication and response time
  e.g., on-site conference registration
- Determine process cycles, i.e., when does each process need to run
  e.g., end-of-month, end-of-project
- Establish processing locations -- identify user locations (and numbers), processor locations, types and numbers, storage devices and storage capacities, connection protocols and traffic volumes
- Distribute data to locations -- simple solution: all in one location; more and more we are moving towards distributed DB solutions
- Distribute software subsystems to locations
- Assign technology -- what hardware, software is going to run where?

Classification of Applications

Span / Type	Operational Support	Decision Support (browsing+analysis)	Real Time
Group/Dept	E.g., regional inventory control	E.g., regional marketing info system	E.g., video conferencing within group
Enterprise	E.g., enterprise-wide cash mgt	E.g., corporate data warehouse	E.g., enterpr-wide video-conference
Inter-Enterprise	E.g., B2B Ecommerce	E.g., DBs for communities of interest	E.g., distributed multimedia over the internet

State of the Market

Span / Type	Operational Support	Decision Support (browsing+analysis)	Real Time
Group/Dept	PC, Windows, OLTP, OO products	COTS (mainly SQL-based	Multimedia technology maturing
Enterprise	ERPs, OLTP over private intranets	ERPs, Web-based products	ERPs, Web-based technologies
Inter-Enterprise	Ecommerce technologies	Web-based technologies	Web-based technologies

- **OLTP --** On-Line Transaction Processing
- **ERPs --** Enterprise Resource Planning systems
- **COTS --** Components Off-The Shelf

Data Management Issues

- Need to identify the amount and type of data persistence needed for the new system:
  - ✓ Is simple file I/O sufficient?
  - ✓ Is a Data Base Management System (DBMS) required?
- A DBMS is typically needed when:
  - ✓ Data is accessed at a fine level of detail,
  - ✓ Sophisticated indexing is required,
  - ✓ There is a need to port data across multiple platforms,
  - ✓ Data needs to be accessible from multiple platforms.

Isolate persistence mechanisms from the application!