Chapter 7;

The computer and Communications, Industries

Lecture 13
TIM 50  Autumn 2012
Thursday  November 8, 2012
Announcement

1. Assignments submitted later than deadline will have 50% points at most.
2. HW 3 and DB assignment 2

DB assignment is due two weeks later.(11/20/2012).

3. The grades for every assignment will be given in eCommons.
4. It's important to check webpage to get the latest information and assignments changes.
5. Business paper group released.
   Submit your business proposal through e-Common

Check class webpage Frequently!
**Review**

**Modularity and Layering**

- Business IT System
- Computer H/W, S/W
- Communication H/W, S/W
- Architecture, Structures
- Sub-Systems
  - Components
  - Modules
- System H/W, S/W
  - Functionality
  - Efficiency
  - Flexibility
  - Interoperability
- Couplings
  - Control
- Information
  - Data
  - Encoding/decoding
  - Data processing
  - Communications
  - Packet, Frame, Encapsulation

- S/W design
  - Problems
  - Solutions = Algorithm
  - Sequences of C.L

- Modularity
  - Layered Architecture
  - Interoperability
  - Interface
  - Abstraction

- Computer
  - Networks
  - OS, Systems
  - Protocols
  - Middleware
  - Applications
  - Data Regenerations
Contents are re-edited using materials from Internet Sites
Software of Complexity

Importance of Design/Architecture (Structure)

- Buying a House
- Marriage
- Get a Job
- Making a Friends

Why S/W Complexity? Complex Systems!

<table>
<thead>
<tr>
<th>S/W</th>
<th>Computer Systems</th>
<th>Communications</th>
<th>Human Systems</th>
<th>Society</th>
</tr>
</thead>
</table>

Solutions Results

Garbage in Garbage Out (GIGO)
Capacity of Individual Human Beings makes the System
What is IT Architecture?

An IT Architecture defines the hardware and software structures of an IT System Solution to a Business problem:

• The **components** that will be reused, developed or purchased

• The **relationships** between components:
  - placement of components on nodes
  - distribution geographically
  - network connections and topology
  - how they will be managed

• How the components interact **dynamically**:
  - interfaces
  - protocols
What is an IT Architect?

The IT Architect defines - architects -- solutions to client business problems through the reasoned application of information technology.

They may also be involved in the integration of a broad variety of products, technologies and services, various systems and applications architectures, and diverse hardware and software components.

Has extensive knowledge of systems, architectures, systems management, networking, network computing and application design techniques.

Is able to identify, evaluate and select the elements of the solution which best meet the needs of the client.

- Provides Vision and Direction
- Defines Scope and Reduces Complexity
- Analyses Required Functionality
- Defines Strategy of Solution
- Defines Rules and Policies
- Makes and Records Formal Decisions
- Bridges IT and Business
- Has end to end responsibility over solution designs
- Assist with providing estimates – Cost, Duration, Skills
IT Architectures are similar to the architectures we find in everyday life.
Complexity

• A system that cannot be understood in all its detail by a single person or small group of people is complex.

• The intricacy of the logic embodied in software
  – suffers no physical limitations
  – complexity is a primary limitation
  – advances allow us to extend that complexity
Goal

- Appreciate the importance of complexity management in networked computing
- Understand better the role of architecture in complexity management
- Examine infrastructure layering in more depth
IT architectures are required to make sure that we create workable solutions that meet business needs.

Architecture assists with translating business requirements into workable and holistic architectures/solutions to suit the business requirements.
There’s rarely one single “correct” solution, each solution is influenced by the current requirements and environment.
Some solutions to complexity

• **Modularity properties**
  – separation of concerns
  – reuse

• **Interoperability through interfaces**
  – abstraction
  – encapsulation
Software Design

– Good design is not accomplished by chance

“The beginning of wisdom for a computer programmer is to recognize the difference between getting a program to work, and getting it right.” [Jackson]

– Fundamental concepts provide the framework for “getting it right”
Phases in the Design Process

- Requirements specification
  - Architectural design
    - System architecture
  - Abstract specification
    - Software specification
  - Interface design
    - Interface specification
  - Component design
    - Component specification
  - Data structure design
    - Data structure specification
  - Algorithm design
    - Algorithm specification

Design activities

Design products
Design Phases

• **Architectural design:** Identify sub-systems.
• **Abstract specification:** Specify sub-systems.
• **Interface design:** Describe sub-system interfaces.
• **Component design:** Decompose sub-systems into components.
• **Data structure design:** Design data structures to hold problem data.
• **Algorithm design:** Design algorithms for problem functions.
Hierarchical Design Structure
Top-down Design

- In principle, top-down design involves starting at the uppermost components in the hierarchy and working down the hierarchy level by level.
- In practice, large systems design is never truly top-down.
- Some branches are designed before others.
- Designers reuse experience (and sometimes components) during the design process.
Software Design

• Design activities having two important parts
  – High-level design
    • Outcome is program structure or software architecture
  – Detailed design
    • The data structure and the algorithms of different modules are designed

Question

How to distinguish a good design from a bad design?
Layered Infrastructure Software
(Infrastructure Software Layering)

Infra ???

Business Enterprise Infrastructure(Infra)

Division and Factory facility
Quality of Employers
IT System Networking
Benefits
Housing/ Environment
Q of L

University Infrastructure(Infra)

Good professor
Building and Lab.
Student Academic Guidance System
Student Life Supporting
Cultural Sites
Architectural Styles

(a) The layered architectural style

(b) The object-based architectural style.
Layer above is a client of the layer below

Each layer provides services to the layer above....

....by utilizing the services of the layer below and adding capability

Layer below as a server to the layer above
We can See, Feel, and Understand!

Information

Representation as data

Information is data with known and consistent structure and interpretation in the context of the current layer.

Representation is a coding of information by data in a form that can be manipulated by a lower layer; the results remain meaningful at the higher layer.

Where, What?
Functional Independence

- **COHESION**
  - the degree to which a module performs one and only one function

- **COUPLING**
  - the degree to which a module is “connected to” other modules in the system
Granularity and Hierarchy

Granularity
The number of modules, the range of functionality

Fine Granularity or a Coarse Granularity
Compromise Functionality and Granularity

Hierarchy

Small modules inside Modules
Hierarchy

Software:
Allows a system to be understood at different

Organization:
Allows a manager to focus on high-level objectives, delegating low-level detail
Information Hiding

- module
- controlled interface

clients

"secret"

- algorithm
- data structure
- details of external interface
- resource allocation policy

a specific design decision
Parts of a module

Module = Interface + Implementation

- What other modules see
- What only the implementer sees
Interface

The Interactions between Modules
  H/W Interfaces
  S/W Interfaces
  Human Interfaces
  Message
  Ingredients of Interface

  Actions
  Parameters and Returns
  Action Menus
  Data Types
  Protocols
Data Used in the Computers

- Integer
- Float/Floating Point
- Character
- String (String of Character)

Data is expressed Name, Type and Value
Integers can be thought of as discrete, equally spaced points on an infinitely long number line. Nonnegative integers (purple) and negative integers (red).
Floating point number

The first programmable computer, the Z3 included floating point arithmetic (replica on display at Deutsches Museum in Munich).

A diagram showing a representation of a floating point number using a mantissa and an exponent.
Good Architecture
S/W

Layering Principles
Special form of Modularity

Application
Middleware
Operating System
Network

Building Construction
Building House
Assembly
Inventory
Supply
Three types of software

- **Components and frameworks:**
  What is in common among applications

- **Infrastructure:**
  Basic services (communication, storage, concurrency, presentation, etc.)
The Layers in a Computing Infrastructure

- Application
- Application Components
- Middleware
- Operating System (O/S)
- Network

Client A

- Application
- Middleware
- Operating System

Client B

- Application
- Middleware
- Operating System

Networks Bus
Data and Information in Layers

Network Infrastructure (H/W and S/W)

Data, Information Storage + Data Communication

Data Processing

Information → Encoding → Data
(Letter, number, Words) (Mapping) (String Of 0, 1 + Standard)

Communication
Protocol Packet, frame, Encapsulates Modem, Gate, Router

Data → Decoding → Information, Applications

Original Information
Data and information

Application
Deals with information

Assumes structure and interpretation

Ignores structure and interpretation

Infrastructure
Deals with data
Storage

Application
Deals with information

Assumes standard data types and SQL = structured query language

Database management system (DBMS)
File system

The infrastructure can provide data management functions
Communication

Application
Deals with information

Assumes standard data types and performs conversions

Distributed object management
Network

The infrastructure can transparently convert representations across platforms
A Package of Data

Data is stored in the format of the File

Data is communicated as a message (Packets, Frames)

Example of Web browser

Web Server

File System

File

Message

Operating System

File Fragmentation

Packets, Frame

Collection of Packets

Assembles

Screen

Web Browser

HTML

Client
Responding to Platform Heterogeneity

**In general**, Heterogeneous Platforms (Different) (Computer, Data Processing Systems)

IBM, MacOS, Main Frame, Unix

Infrastructures can understand;
The Structures + Interpretation of the Data from Different Systems

1. Functions in a Individual Systems
2. Convert to Standard Data Type, then Regenerate Originals
Middleware System

Additional layer on the top of NOS implementing general-purpose services. Better transparency.
Middleware

• Software that manages and supports the different components of a distributed system.
  In essence, it sits in the *middle* of the system
• Middleware is usually off-the-shelf rather than specially written software
• Examples
  – Transaction processing monitors
  – Data converters
  – Communication controllers
Layered application architecture

• Presentation layer
  – Concerned with presenting the results of a computation to system users and with collecting user inputs

• Application processing layer
  – Concerned with providing application specific functionality e.g., in a banking system, banking functions such as open account, close account, etc.

• Data management layer
  – Concerned with managing the system databases
Communications middleware components

If the same application is to be used with a different database and network, the application’s routines must be rewritten for the new database and network protocols.

- This is where middleware comes in handy.

The use of middleware yields,

**Network independence** by allowing front-end application to access data without regard to the network protocols.

**Database server independence** by allowing front-end user to request from multiple database servers with different SQL codes.
Database middleware components

- Middleware software is divided into three main components:
  - **Applications programming interface (API)**:
    Middleware allows the client process to be database-server-independent.
  
  - **Database translator**:
    Translate the SQL requests into the specific database server syntax.
  
  - **Network translator**:
    Manages the network communications protocols.
Database middleware components

Client front-end

API
DATABASE TRANSLATOR
NETWORK TRANSLATOR

Database middleware

Network protocol
Horizontal structure in layers
A spanning layer is ubiquitous and hides the layers below.
More on Good Architecture

Abstraction

Encapsulations

Flexibility
Abstraction

• A property of well-designed interfaces to modules
• Hide detail, displaying only what is necessary
• Simplify, displaying only what is meaningful to the outside
• Important for complexity management
Encapsulation

• Module implementation details (anything not explicit at interface) should be inaccessible from the outside
  – So other modules cannot become inadvertently dependent on implementation
  – In the case of components, for proprietary or security reasons
Protocols

• A defined sequence of actions between/among two or more subsystems required to achieve some higher-level functionality

• Interface specification focuses on actions (including formats of parameters and returns) and protocols
Summary of modularity

• **Divide and conquer**: decomposition of the system into modules with well-defined functional groupings

• **Separation of concerns**: great dependency internally, little dependency externally

• **Abstraction**: hide detail and simplify

• **Encapsulation**: make internal implementation inaccessible

• **Reusability**: meet generalized needs, configurable
Understand better

how layering is used in the infrastructure
how it contains complexity
how it coordinates suppliers
how it allows new capabilities to be added incrementally
• Given a problem to solve
  – Analyze it
  – Synthesize a solution
• Understand that requirements may change
• Must view quality from several different perspectives
• Use fundamental software engineering concepts (e.g., abstractions and measurements)
• Keep system boundary in mind
The systems engineering and architecture discipline

Wikipedia: Systems Engineering

- A branch of engineering whose responsibility is creating and executing an interdisciplinary process to ensure that customer and stakeholder’s needs are satisfied in a high quality, trustworthy, cost efficient and schedule compliant manner throughout a system’s entire life cycle, from development to operation to disposal.
- This process is usually comprised of the following seven tasks:
  - State the problem
  - Investigate alternatives
  - Model the system
  - Integrate
  - Launch the system
  - Assess performance
  - Re-evaluate


Systems engineering and architecture discipline

- The systems engineering and architecture discipline is a full lifecycle and interdisplinary approach and a means to enable the implementation of successful systems.
- Systems engineering and architecture discipline focuses on defining customer needs and required functionality early in solution design and the development cycle, eliciting requirements, and then proceeding with design synthesis and system validation while considering the complete problem.
- This discipline considers both the business and the technical needs of the customers and stakeholders with the goal of providing quality solutions or products that meets user needs.
The Computer and Communications Industries

Computer/Networks/Data Communications

Architects/Layered H/W, S/W Components/Modules

Stovepipe Structured Systems
Integrated Infrastructures (Networked/Distributed)

standardizations
Components

- Components provide a service without regard to where the component is executing or its programming language
  - A component is an independent executable entity that can be made up of one or more executable objects;
  - The component interface is published and all interactions are through the published interface;
Component as a service provider

- The component is an independent, executable entity. It does not have to be compiled before it is used with other components.
- The services offered by a component are made available through an interface and all component interactions take place through that interface.
Architecture provides a set of laws/boundaries for solutions, to ensure that all components of the solution will work together.

Catalogs, provide *detailed specifications* of the available parts together with a variety of alternative products implementing those specifications; *plus recommendations on how to use these parts* (reference architectures).

Solutions to many different requirements, all *built from the same common collection of parts*, each designed according to accepted standard reference architectures.
IT Architecture must consider the views of different users and their needs

Each view has a different architectural representation, just as a electrical wiring diagram is different from the interior designer’s “picture”.

- What processes, information, roles and locations must be addressed?
- What information needs to be made available, to whom, and how?
- What are the data elements, where stored, and how are they accessed?
- What infrastructure do we need to provide the required Service levels?
- What are the security and privacy requirements for the infrastructure and applications?
- How do applications support the required functionality?
- How do we integrate with external systems and databases?
- What types of content, how is it maintained, published, and distributed?
- How do you manage and administer the infrastructure and the application?
IT Architectures behind everyday life - Google
IT Architecture behind everyday life – Mobile/Cellular Network
There are many specialised skills required to create a building architecture and the same applies to an IT solution.

The different types of architects

- Infrastructure (hardware)
- Data
- Application (software)
- Integration (data, application)
- Enterprise (the holistic picture)
- Business architect (business process)
- Systems Engineering and Architecture
Components with Workflow Overview
Product Characteristics

- **Components:**
  - *Case* - The box all the parts (except monitor, keyboard, mouse, and printer) are stored in
  - *Mother Board* - The main printed circuit board in the computer, which the CPU (see below) plugs into
  - *CPU (Central Processing Unit)* – The CPU is the actual "brains" of the computer
  - **RAM (Random Access Memory)** - Like pieces of scratch paper that information is temporarily stored on ONLY WHILE you are actually working on the computer.
  - *Disk Controller* – The Disk Controller allows your computer to interact with your disk drive storage devices
  - *Hard Disk Drive* - A STORAGE device, NOT MEMORY! The Hard Disk Drive is like a filing cabinet - no more, no less. Retrieval is faster, and finding things is usually easier, but it is still just a filing cabinet.
  - *Video Display Adapter* - Unlike your eyes, it can ONLY OUTPUT the computer information in the form of a video signal that is human readable (via the monitor).
  - *Monitor* - The actual display you see the words, pictures, and data on. There are two main types: analog and digital.
  - *Input Device* - Keyboard, Mouse, Digitizer, Scanner, Pen, Digital Camera, etc.
  - *Modem* - A device that hooks your computer up to the telephone line
Computer Parts

• There are many parts that work together to make a computer work.
Hardware

- Physical parts of the computer, including processor and memory chips, input/output devices, tapes, disks, modems, cable, etc.
CPU

• The Central Processing Unit
Processor

• The CPU – The chip or chips that interpret and execute program instructions and manage the functions of input, output, and storage devices.
Computer Case

- Contains the major components of the computer. It helps protect them.
Front of the Computer Case

- Built-in Handle
- Power Switch
- 5.25" Drive Bays
- Floppy Drive Bay
- Front USB Ports
Inside the Computer Case

- Power Supply
- 5.25” Drive Bays
- Expansion Slots
- 3.5” Drive Bays
Monitor

• A display screen to provide “output” to the user. It is where you view the information you are working on.
Video Card

• Connects the computer to the monitor. It is a circuit board attached to the motherboard that contains the memory and other circuitry necessary to send information to the monitor for display on screen.
Keyboard

• Used to enter information into the computer and for giving commands.
Mouse

- An input device operated by rolling its ball across a flat surface. The mouse is used to control the on-screen pointer by pointing and clicking, double-clicking, or dragging objects on the screen.
Touchpad

- A pressure-sensitive and motion sensitive device used in place of a mouse.
CD Rom Drive

- The drive that plays CDs and reads data that has been stored on the CD.
CD

• Compact Disk – A type of optical storage device.
Floppy Disk Drive

• A device that holds a removable floppy disk when in use; read/write heads read and write data to the diskette.
Hard Disk

• Magnetic storage device in the computer.
RAM

• Random Access Memory
RAM is a computer’s temporary memory, which exists as chips on the motherboard near the CPU. It stores data or programs while they are being used and requires power.
Printer

- An output device that produces a hard copy on paper. It gives information to the user in printed form.
Barcode Reader

- An input device that converts a pattern of printed bars into a number that a computer can read. They are often used by businesses to quickly input price and product information.
Scanner

• A device that allows pictures to be placed into a computer.
Microphone

- Allows the user to record sounds as input to their computer.
Speakers

- Used to generate or reproduce voice, music, and other sounds.
Sound Card

- Connects the speakers and microphone to the computer.
Modem

- The place where the computer is connected to the phone line.
Network Card

• A circuit board that connects the computer to the rest of the network usually using special cables.
Software

- Programs that tell the computer what to do. It provides instructions that the CPU will need to carry out.
DOS

- Disk Operating System
  This software connects the hardware with the programs you want to run.

MS-DOS:

Microsoft DOS (Disk Operating System) is a command line user interface. MS-DOS 1.0 was released in 1981 for IBM computers.
Example of MS-DOS
Windows

- A family of operating systems developed and produced by Microsoft Corp. It provides a software graphical user interface (GUI) used on IBM and compatible computers.
Example of Windows (GUI)
Network Hardware
Transmission Direction

- simplex: One direction only
Half Duplex Transmission

half duplex: Both directions but only one direction at a time.
Full Duplex Transmission

full duplex:
send and receive both directions at once
Network Protocol

Ethernet

- Developed at Xerox in 1976.
- First protocol approved as an industry standard protocol 1983
- LAN protocol used on bus and star
- Most popular LAN protocol
- Inexpensive
Carrier Sense Multiple Access/Collision Detection (CSMA/CD)

- Used on bus networks to avoid data collisions.
Data Communication Protocol
TCP/IP

• Developed in 1973 for use on the ARPANET which was a defense force research network.

- Adopted in 1983 as the Internet standard. All hosts on the Internet are required to use TCP/IP.

- Allows transfer of data using packet switching

Graham Betts
LANs Vs WANs

- **LAN** is “local Area network” which is a network confined to a small geographic area which is a building or a group of buildings.
- **WAN** is “wide area network” which is a network spread over a large geographic area. The largest WAN is the internet.
Examples of LANS

3 different types of LANS are:

– Ring
– Bus
– Star
NETWORKS: categorized by size

LAN – a network that connects computers in a limited geographical area.

MAN – a backbone that connects LANs in a metropolitan area such as a city and handles the bulk of communications activity across that region.

WAN – covers a large geographical area such as a city or country. Communication channels include telephone lines, Microwave, satellites, etc.
NETWORK TOPOLOGIES
(categorizing by shape)
Large networks can be separated into two or more smaller networks using a bridge. This is done to increase speed and efficiency. This type of network is called a segmented LAN and has largely been superseded by the use of switches which can transfer data straight to a computer and thus avoid bottleneck jams which bridges were designed to fix.
Gateway

Often used to connect a LAN with a WAN. Gateways join two or more different networks together.
## Characteristics of The Internet, Intranet, and Extranet

<table>
<thead>
<tr>
<th>FOCUS</th>
<th>TYPE OF INFORMATION</th>
<th>USERS</th>
<th>ACCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Internet</td>
<td>External communications</td>
<td>General public</td>
<td>Any user with an Internet connection/public and not restricted</td>
</tr>
<tr>
<td>The Intranet</td>
<td>Internal</td>
<td>Employees</td>
<td>Authorized employees/private and restriction</td>
</tr>
<tr>
<td>The Extranet</td>
<td>External</td>
<td>Business partners, customers, suppliers</td>
<td>Authorized business partners/private and restriction</td>
</tr>
</tbody>
</table>
Transmission Media

More on internet

• twisted pair – telephone cable
• coaxial cable – Thick black cable used for higher bandwidth communications than twisted pair (i.e. Optus cable)
• fibre optic – data transferred through pulses of light. Extremely fast.
• Non cable methods such as satellite, microwave, wireless and bluetooth
Network Hardware

More on Internet

**SERVERS**: Help to manage the network and the resources of that network. On larger networks servers commonly have specialised tasks such as: **File Servers**: stores and manages files, **Print Servers**: manages printers and print jobs, **Mail Server**: Manages email, **Web Server**: manages web access.

**Routers**: connects multiple networks and are protocol independent. can be used in place of a switch or bridge.

**Switches**: smart hubs which transmit packets to the destination port only

**Hubs**: like double adapters /power boards in the home except instead of plugging in extension cords we are plugging in computers to allow them to communicate.
Other Information Processes in Communication Systems

Collecting: phone as collection device with voice mail, EFTPOS terminal as a collection device for electronic banking

processing: sending of attachments with e-mail, encoding and decoding methods, including: analog data to analog signal, digital data to analog signal, digital data to digital signal, analog data to digital signal, client-server architecture: the client controls the user interface and the application logic server controls access to the database
Collecting: The following are collection devices: ATMs for internet banking, EFTPOS for stores, microphone and video camera for video conferencing. Data can be analog or digital.
Processing

• **Processing**: Is the manipulation or changing the data into a more useable format. The processing may include changing the appearance of the data, the file type or storage options.
Displaying

Displaying: How the information is made available for the user to see
Suppliers/Providers/Customers for Networked Computing Related Product /Services

- Information contents
- Component Supplies, custom subsystem
- Developers, System Integrators
- S/W product
- End-user
- Specific vertical industries Infrastructure
- On-line merchants
- On-line service providers
- Cyberspace consumers
Properties of Products and services

Types of Information Goods
Types of Software Goods
The software products
The customer-developed software application
Infrastructure S/W bulks
Unbundled infrastructure Systems
S/W Custom Developers
Computer/Design Service
What is System Integration?

INCOSE SE Handbook:

...establishment of system interfaces, internal and external...
...emphasis on risk management and continuous verification...

The process of putting a system together, with techniques to ensure all the parts work as a whole.
Integration is Hard

Generally, the main contractor for the project is responsible for systems integration.

The sub-contractors will usually be part of the integration team.

Integration is one of the most costly and time-consuming activities in the systems engineering process.

For large and complex systems, up to 40% of the development effort may be used in this activity, mostly in system testing.
What is Interoperability?

“Ability of systems, units, or forces to provide services to and accept services from other systems, units, or forces and to use the services so exchanged to enable them to operate effectively together.”

*(JCS Pub 1)*
System-of-Systems

CJCSI 3170.01D

System-of-Systems.
A set or arrangement of interdependent systems that are related or connected, to provide a given capability. The loss of any part of the system will degrade the performance or capabilities of the whole.

Key Points

- Arrangement of systems
- Related or connected systems
- Provide a given capability

- Loss of part
- Degrades the whole

One Function Concept

Limited Redundancy
A Typical Integration Example

1. Collect Hardware Components
2. Integrate Hardware Platform
3. Integrate Software on Target Hardware
4. Collect Software Components

- Test System
  - Interfaces
  - Configurations
  - Stress
  - Factory Acceptance

- User Acceptance Test
- Resolve Issues

Human Systems Integration
# Integration Trends

## Integrated Modular Architecture

| Each supplier generally has proprietary hardware (LRU) increasing cost of supply / repair chain and aircraft weight |
| All software in a LRU/card must be developed to the same safety level even, if this is not strictly necessary, and is dedicated to that LRU |
| If the hardware platform changes the whole product needs to re-verified by licensing authority |
Integration Planning
IT Industry Activities

Scope of Competitive Rivalry

- The competition within the PC industry is extraordinarily cutthroat. The top companies consist of Dell, HP, Apple, Gateway and Sony.

- **A few factors of competitive edge:**
  - New technology
  - Custom built PCs
  - Reliability
  - Customer Service
Technology/Innovation

- Technologies as well as innovation are advancing every year, therefore making the industry fiercer.
- Ten years from now chances are there will be few computers in home. Instead, people will be wearing computers -- implanted, for example, in eyeglasses, with the retina as the screen -- according to IT pioneer and futurist Raymond Kurzweil.
Competitive Position of Major Competitors

- Dell
- Hewlett-Packard
- Apple
- Gateway
- 4B
- 3B
- 2B
- 1B
- 25B
- 50B
- 75B
- 100B

Net Income
Threat of New Entrants/Stakeholders

TNE: The chances of a new PC vendor entering into the market and gaining significant market share is pretty slim, entry barriers are higher now than they every have been. Two of those entry barriers that are making entry into the PC market so tough are cost and distribution.

Concentrated: it is estimated by analysts that in the near future the top 5 vendors may control 70% of the global personal computer market.

Stakeholders: Environmentalists and E-waste are the major stakeholders for the PC industry; the dumping of high-tech trash like computers in landfills.

Hewlett-Packard, IBM and Dell have recently started up a computer recycle program, recycling in all computer brands for a small fee.
## Substitute Product

<table>
<thead>
<tr>
<th>Functionality</th>
<th>PC/Notebook</th>
<th>Cell Phone</th>
<th>Handheld devices</th>
</tr>
</thead>
<tbody>
<tr>
<td>PowerPoint</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Camera</td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Mp3 player</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Spreadsheets</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Word processor</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>GPS navigation</td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Email</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
Key Success Factors

- PC Industry Services
- Technology Services
- Consulting and Integration
- Managed Services
The Changing Industry Structure
DISTRIBUTED STRUCTURES

**VOCABULARY**

**Tightly coupled systems**  
Same clock, usually shared memory. Multiprocessors. Communication is via this shared memory.

**Loosely coupled systems**  
Different clock, use communication links. Distributed systems.

- sites = nodes = computers = machines = hosts
- Local  
The resources on your "home" host.
- Remote  
The resources NOT on your "home" host.
- Server  
A host at a site that has a resource used by a Client.
Advantages of distributed systems:

- Site A:
  - Server
  - Communication

- Site B:
  - Client

- Site C:
  - Resources
NETWORK STRUCTURES

LOCAL AREA NETWORKS (LAN):

- Designed to cover small geographical area.
- Multiaccess bus, ring or star network.
- Speed around 1 gigabit / second or higher.
- Broadcast is fast and cheap.
- Usually workstations or personal computers with few mainframes.

WIDE AREA NETWORK (WAN):

- Links geographically separated sites.
- Point to point connections over long-haul lines (often leased from a phone company.)
- Speed around 1 megabits / second. (T1 is 1.544 megabits/second.)
- T-3 - 43.232 megabits per second (28 T-1s) Ave. cost $4,000.-$16,000./mo. (2011)
- Broadcast usually requires multiple messages.
- Nodes usually contain a high percentage of mainframes.
The communication network is partitioned into the following multiple layers:

**Design Structure**

- Application layer
- Presentation layer
- Session layer
- Transport layer
- Network layer
- Link layer
- Physical layer

**Current Operations**

- End-to-end message transfer (connection management, error control, fragmentation, flow control)
- Network routing, addressing, call set-up and clearing
- Data-link control (framing, data transparency, error control)
- Mechanical and electrical network-interface connections
- Physical connection to network termination equipment

End-user application process
- Distributed information services
- File transfer, access, and management; document and message interchange, job transfer and manipulation
- Syntax-independent message interchange service
- Transfer syntax negotiation; data representation transformations
- Dialog and synchronization control for application entities
- Network-independent message interchange service

Data-communication network
# NETWORK STRUCTURES

<table>
<thead>
<tr>
<th>Layer</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Physical layer</strong></td>
<td>Handles the mechanical and electrical details of the physical transmission of a bit stream.</td>
</tr>
<tr>
<td><strong>Data-link layer</strong></td>
<td>Handles the frames, or fixed-length parts of packets, including any error detection and recovery that occurred in the physical layer.</td>
</tr>
<tr>
<td><strong>Network layer</strong></td>
<td>Provides connections and routing of packets in the communication network. Includes handling the address of outgoing packets, decoding the address of incoming packets, and maintaining routing information for proper response to changing load levels.</td>
</tr>
<tr>
<td><strong>Transport layer</strong></td>
<td>Responsible for low-level network access and for message transfer between clients. Includes partitioning messages into packets, maintaining packet order, controlling flow, and generating physical addresses.</td>
</tr>
</tbody>
</table>
**NETWORK STRUCTURES**

**Presentation layer**  Resolves the differences in formats among the various sites in the network, including character conversions, and half duplex/full duplex (echoing).

**Application layer**  Interacts directly with the users. Deals with file transfer, remote-login protocols and electronic mail, as well as schemas for distributed databases.
How this is really implemented can be seen in this figure:

end-user application process

- file transfer protocol, FTP
- remote terminal protocol, TELNET
- simple mail transfer protocol, SMTP
- name server protocol, NSP
- simple network management protocol, SNMP

TCP
UDP
IP
IEEE802.X/X.25

LAN/WAN

TCP = transmission control protocol
UDP = user datagram protocol
IP = internet protocol
The Role of Architecture from stovepipe to Integrated Infrastructure

Less Vertical Integration and more diversifications

Venture capital and stand-ups
Net-Centric Operations
A Fundamental Shift

Platform-Centric

Net-Centric

Traditional stove-pipe approach v. Fused information available on the Net

Source: Margaret Myers, Principal Deputy OSD DCIO
System-Centric, “Stovepipe” Architecture Problems

- Pre-web, client/server, terminal/mainframe paradigm
  - Content is typically application specific and relies on specific systems and platforms to provide access, access control, and presentation
  - Content is often not visible to the web, i.e. “Deep Web” problem
- Content lacks flexibility and extensibility. Difficult to do versioning, dynamic creation, appropriate disseminations, and support intra/inter-organizational reuse or value-adding
- Content quickly becomes unauthoritative as it is downloaded and reused
  - Outdated
  - Insecure
- Architecture lacks coherent preservation strategy
What are the Characteristics of a Net-Centric Information Space?

- Content is:
  - Persistent
  - Visible
  - Authoritative
  - Extensible
  - Reusable
  - Interoperable across platforms

- Content supports:
  - Dynamic relationships
  - Value adding
Digital Object Architecture

- Network paradigm
  - Content is *not* application specific and does not rely on specific *systems and platforms* to provide access, access control, and presentation
  - Content can easily be made visible to the web
- Content is flexible and extensible
  - Lends itself to versioning, dynamic creation, appropriate disseminations, and intra/inter-organizational reuse
  - Supports the establishment of content relationships
  - Is networkable across system/organizational boundaries
  - Content remains authoritative as it is accessed and reused
  - Supports coherent preservation strategies
Digital Object Architecture Components

Digital Objects
- Persistently identified
- Self contained
  - Self described
  - Self aware
- Integral access control
- Extensible

Repositories
- Contain digital objects
- Are, themselves, digital objects
- Repository Access Protocol (RAP)
- Platform independent

Handle System
- Persistent identification
- Distributed Architecture

Resource Discovery
- Search engines
- Databases
- Digital information
- Digital objects
Digital Objects

Content-centric, instead of, System-centric
Digital Object Chaining

Handle - German Summary of Hamlet

Content Type Disseminators

Handle
Type - Text
Servlet
Attachments

Data Elements

Handle
Type - Summary
Servlet
Attachments

Data Elements

Handle
Type - German tr.
Servlet
Attachments

Data Elements

Hamlet - Text → Hamlet - Summary → Hamlet - German Summary
Repositories

Also include content disseminations

Client

Repository1

Repository2

RAP

RAP
Digital Object Architecture Business Case

- Provides useful, authoritative, visible, and dynamic access to information
  - Encourages the innovative, value-added, dynamic and, therefore, effective use of digital information
  - Provides global visibility/coordination over information
  - Controls access to information
  - Reuses generated content, techniques, and disseminations
  - Preserves information functionality
  - Discourages generation of multiple, and potentially different, versions of information

- Provides interoperability and platform independence
- Provides greater security
  - Supports real-time access control changes
  - Supports partial dissemination
System-Centric, “Stovepipe” Architecture Problems

• Pre-web, client/server, terminal/mainframe paradigm
  – Content is typically application specific and relies on specific systems and platforms to provide access, access control, and presentation
  – Content is often not visible to the web, i.e. “Deep Web” problem

• Content lacks flexibility and extensibility. Difficult to do versioning, dynamic creation, appropriate disseminations, and support intra/inter-organizational reuse or value-adding

• Content quickly becomes unauthoritative as it is downloaded and reused
  – Outdated
  – Insecure

• Architecture lacks coherent preservation strategy
What are the Characteristics of a Net-Centric Information Space?

• Content is:
  – **Persistent**
  – Visible
  – Authoritative
  – Extensible
  – Reusable
  – Interoperable across platforms

• Content supports:
  – Dynamic relationships
  – Value adding
Digital Object Architecture

• Network paradigm
  – Content is not application specific and does not rely on specific systems and platforms to provide access, access control, and presentation
  – Content can easily be made visible to the web
• Content is flexible and extensible
  – Lends itself to versioning, dynamic creation, appropriate disseminations, and intra/inter-organizational reuse
  – Supports the establishment of content relationships
  – Is networkable across system/organizational boundaries
  – Content remains authoritative as it is accessed and reused
  – Supports coherent preservation strategies
Digital Object Architecture Components

Digital Objects
- Persistently identified
- Self contained
  - Self described
  - Self aware
- Integral access control
- Extensible

Repositories
- Contain digital objects
- Are, themselves, digital objects
- Repository Access Protocol (RAP)
- Platform independent

Handle System
- Persistent identification
- Distributed Architecture

Resource Discovery
- Search engines
- Databases
- Digital information
- Digital objects
Digital Objects

Content-centric, instead of, System-centric
Digital Object Example: Federal Management Regulation

Content Type Disseminators

Version disseminations

Handle - FMR

Data Elements

Handle
Type - FMR
Servlet
Attachments

Version 1

Data Elements

Handle
Type - FMR
Servlet
Attachments

Version 2

Data Elements

Handle
Type – FMR
Servlet
Attachments

Version 3

Corporation for National Research Initiatives
Digital Object Chaining

Content Type Disseminators

Handle - German Summary of Hamlet

Data Elements

Handle
Type - Text
Servlet
Attachments

Data Elements
Handle
Type - Summary
Servlet
Attachments

Data Elements
Handle
Type – German tr.
Servlet
Attachments

Hamlet - Text  ➔  Hamlet - Summary  ➔  Hamlet - German Summary

Corporation for National Research Initiatives
Repositories

Also include content disseminations
Digital Object Architecture Business Case

- Provides useful, authoritative, visible, and dynamic access to information
  - Encourages the innovative, value-added, dynamic and, therefore, effective use of digital information
  - Provides global visibility/coordination over information
  - Controls access to information
  - Reuses generated content, techniques, and disseminations
  - Preserves information functionality
  - Discourages generation of multiple, and potentially different, versions of information

- Provides interoperability and platform independence

- Provides greater security
  - Supports real-time access control changes
  - Supports partial dissemination
What’s Been Done

• Handles:
  – Established Handles infrastructure - CNRI
  – Established DOIs, registry agencies, and transfer mechanisms for the commercial publishing industry -- CNRI and DOI Foundation
  – Implemented the DOD Handles Service and Handles Registry for DTIC held unclassified/unlimited technical reports -- DTIC

• Digital Object Architecture:
  – Implemented a prototype Digital Object Repository, i.e. Defense Virtual Library, for technical reports, photographs, sound files, and moving images -- DTIC and CNRI
  – Successfully tested repository interoperability -- CNRI and Cornell University
## Evolution of Enterprise Computing

<table>
<thead>
<tr>
<th>Time Period</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Early 1990s</strong></td>
<td>Automated Enterprise</td>
</tr>
<tr>
<td></td>
<td>· Enterprise Applications</td>
</tr>
<tr>
<td></td>
<td>· Real-Time, Integrated Enterprise</td>
</tr>
<tr>
<td><strong>Emerging</strong></td>
<td>Dynamic, Adaptive Enterprise</td>
</tr>
<tr>
<td></td>
<td>· Real-time analysis and closed loop feedback</td>
</tr>
<tr>
<td><strong>Now-to-Five Years Out</strong></td>
<td>Adaptable, Self-Organizing, Collaborative Supply Networks</td>
</tr>
<tr>
<td></td>
<td>· Automated capabilities via collaborating agents (cogents)</td>
</tr>
<tr>
<td><strong>Beyond Five Years Out</strong></td>
<td></td>
</tr>
</tbody>
</table>
Emerging Middleware Industry

- System Integrators
- Packaged Applications
- Valued-Added Resellers
- Computer Technology
  - Oracle RDMS
  - IBM MQ Series
  - Microsoft MSMQ
  - Inprise Visi-Broker
  - Iona Orbix

Computer Platforms
# Emerging Middleware Industry

<table>
<thead>
<tr>
<th>Service Type</th>
<th>Vendor(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Management</td>
<td>Tivoli</td>
</tr>
<tr>
<td>Application Integration</td>
<td>Crossworld</td>
</tr>
<tr>
<td>Push</td>
<td>Backweb</td>
</tr>
<tr>
<td>Workflow</td>
<td>Vitria</td>
</tr>
<tr>
<td>Message Broker</td>
<td>Neon, Mercator</td>
</tr>
<tr>
<td>Transaction Monitor</td>
<td>IBM MQ</td>
</tr>
<tr>
<td>Object-Based</td>
<td>Tuxedo</td>
</tr>
<tr>
<td>Message-Based Middleware</td>
<td>COM/DCOM, Corba, EJB</td>
</tr>
<tr>
<td>Reliable Multicast</td>
<td>Cisco</td>
</tr>
</tbody>
</table>
Elements of Middleware

- Web applications infrastructure
  - Infrastructure for enabling distributed web- and Internet-related applications
- Multi-agent systems
  - Infrastructure for enabling systems of cooperating independent agents
- Distributed object/component systems
  - Infrastructure for enabling interactions among distributed objects and components (including three major approaches—DCOM, CORBA, and Java)
- Message-oriented middleware (MOM)
  - Infrastructure for message passing among distributed computing elements
- Distributed database applications infrastructure
  - Infrastructure for distributed database applications
Relevent Commercial Technologies

- **Web Technologies**
  - DHTML, XML, DOM, HTTP-NG

- **Agent Technology**
  - Multiagent Systems

- **Collaborative Apps**
  - Notes

- **Distributed App Building Blocks**
  - RPC & Java RMI

- **Message-Oriented Middleware (MOM)**
  - IBM’s MQ series, Tibco’s “information bus,” Lotus Notes/Domino

- **Distributed Object /Component-based Systems**
  - Microsoft’s COM, COM+, DCOM; OMG’s CORBA; and Java (Java Beans, Enterprise Java Beans, Javaspaces, Jini, etc.)

- **Distributed DB Technologies**
  - XACT servers, ODBC & JDBC

Areas of Concentration
# Architectural Framework

<table>
<thead>
<tr>
<th>Tools</th>
<th>UI/Navigation</th>
<th>Distributed OS Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTML/Scripting Authoring</td>
<td>Basic html</td>
<td>Management</td>
</tr>
<tr>
<td>Rapid Applications Development</td>
<td>Dyn html Forms Native html</td>
<td>Directory</td>
</tr>
<tr>
<td>Component Creation</td>
<td>Business Process</td>
<td>Security</td>
</tr>
<tr>
<td>Team Development</td>
<td>Integrated Storage</td>
<td>Networking</td>
</tr>
<tr>
<td></td>
<td>File Database Mail Other Stores</td>
<td>Base Services</td>
</tr>
<tr>
<td></td>
<td>Web Svr Transactions Queuing</td>
<td></td>
</tr>
</tbody>
</table>
Emerging Generic Middleware Architecture

Client
- User Interface
- Forms

Enterprise
- Web Server
- Business Logic
- XACT/DBMS
- Computer Storage

Legacy
- Packaged Applications (SAP, Baan, PeopleSoft)
- Old Apps & Data Sets
Distributed Systems Model
Equipment Convergences

• Computing/Communication
• Computing/Communication/Video
• Computing/communication/Media
• ....... For the Future!
Standardization
Reference Model and Interface & Industry organization and standardization Processes
Who controls and who enforces standards
Why open standardization Has Down sides too.
The main criteria for international standardization are:

- **Improvement in universal technical communication and mutual understanding;**
- **Facilitation of international exchange of goods and services;**
- **Removal of technical barriers to trade;**
- **Transfer of technology.**
Benefits of Standardization

- Uniform terminology is created
- Sizes and dimensions are co-ordinated and adapted
- Variety is reduced
- Function requirements and characteristics are specified
- Unambiguous testing methods are established
For **product standards** the benefits may be broadly summarized under the headings variety reduction, interchangeability, and availability:

the effects of **variety reduction** are well known and can mostly be assessed in terms of hard cash, taking due account of the additional cost reduction effects on associated parts and operations;

**interchangeability** as a result of standardization leads to higher productivity and lower manufacturing costs. These benefits are relayed to the users as a result of increased competition. Interchangeability is also very important to erection, installation, maintenance, and repairs;

easy **availability** implies that an adequate number of varieties are always in stock. This means shorter lead times and less capital tied up on the user side.
Benefits of Standardization

- Products are in different places
- Brands are different
- New products unknown to you
Customers have different tastes, different pockets, and buy every time for different reasons.
Benefits of Standardization

Users and Consumers

**USERS:** Buy to add value to the product and make profit

**CONSUMERS:** Buy to consume the product and get satisfaction at a reasonable price.
Benefits of Standardization

Is there enough offer of standards?
Is there enough demand of standards?
Are there too many standards?
Are they sufficiently employed?
Benefits of Standardization

The Universe of the Standardization

Level

International
Regional
National
Sectoral
Company

Activity

Terminology
Technical Specifications
Sampling and control
Testing and analysing
Reduction of the variety
Grading
Codes of Good Practices

Field

Management
Electrical
Food Industry
Agriculture
Forestry
Textile Industry
Chemical Products
Information Technology
Education and Training
Benefits of Standardization

Relationship between standardization at company, national, and international levels

*It should always be remembered that it is on the company level that the actual implementation of standards takes place.*
Benefits of Standardization

To industries:

— market research

— development;
  — engineering;
  — production engineering;
  — tooling;
  — estimating;
  — ordering;
  — works order planning;
  — purchasing;
  — component production;
  — environment;
  — assembly;
  — environment;
  — inspection;
  — storing, warehousing;
  — marketing;

— distribution;
— transport;
— installation;
— final inspection;
— operation and use;
— maintenance;
— repairs;
— extensions;
— effects from
— effects on
— dismantling, recycling;
— destruction.

— Support to management systems
Benefits of Standardization

To consumers:

• A source of information
• The application of the performance approach
• Comparative analysis
• Source for complaints
• More balanced technical regulations
Benefits of Standardization

According to the environment, emphasize:

• Economic benefits
• Technical Benefits
• Social benefits
• Political benefits
• Management benefits
Open Issues

Industry Organization
The Best Standardization Process
Review of Class
The Computer and Communications Industries

Computer/Networks/Data Communications

Architects/Layered H/W, S/W Components/Modules

Stovepipe Structured Systems
Integrated Infrastructures (Networked/Distributed)

standardizations