NETWORK 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**.

Network Operating Systems
- The users are aware of the physical structure of the network.
- Each site has its own OS and some protocol (i.e. FTP) provides an interface to those OS.
- Users must know machine and directory structure in order to find a file.

Distributed Operating Systems
- The users are UNaware of the physical structure of the network.
- Data and process usage appears seamless.

Clusters
- The hardware on which distributed systems run. A current buzzword.
- It allows more compute power, compared to a mainframe, by running on many inexpensive small machines.

Chapter 16 talks in great deal about distributed systems as a whole; meanwhile we'll discuss the components of these systems.

MOTIVATION

Advantages of distributed systems:

Resource Sharing

- Items such as printers, specialized processors, disk farms, files can be shared among various sites.

Computation Speedup

- Load balancing - dividing up all the work evenly between sites. Making use of parallelism.

Reliability

- Redundancy. With proper configuration, when one site goes down, the others can continue. But this doesn't happen automatically.

Communications

- Messaging can be accomplished very efficiently.
- Messages between nodes are akin to IPCs within a UniProcessor.
- Easier to talk/mail between users.

TOPOLOGY

Methods of connecting sites together can be evaluated as follows:

- **Basic cost:** This is the price of wiring, which is proportional to the number of connections.

- **Communication cost:** The time required to send a message. This is proportional to the amount of wire and the number of nodes traversed.

- **Reliability**: If one site fails, can others continue to communicate.

Let's look at a number of connection mechanisms using these criteria:

FULLY CONNECTED

- All sites are connected to all other sites. **<<< FIGURE 15.2 >>>**

- Expensive(proportional to N squared), fast communication, reliable.

PARTIALLY CONNECTED

- Direct links exist between some, but not all, sites. **<<< FIGURE 15.3 >>>**
- Cheaper, slower, an error can partition system.

HIERARCHICAL

- Links are formed in a tree structure. **<<< FIGURE 15.4 >>>**
- Cheaper than partially connected; slower; children of failed components can't communicate.

STAR

- All sites connected through a central site. **<<< FIGURE 15.5 >>>**
- Basic cost low; bottleneck and reliability are low at hub.

RING

- Uni or bi-directional, single, double link. **<<< FIGURE 15.6 >>>**
- Cost is linear with number of sites; communication cost is high; failure of any site partitions ring.

MULTIACCESS BUS

- Nodes hang off a ring rather than being part of it. **<<< FIGURE 15.7 >>>**
- Cost is linear; communication cost is low; site failure doesn't affect partitioning.

NETWORK TYPES

LOCAL AREA NETWORKS (LAN):

- Designed to cover small geographical area.
- Multiaccess bus, ring or star network.
- Speed around 10 megabits / second or higher. (Gigabits networks are on the way!)
- Broadcast is fast and cheap.
- Nodes are 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.)
- Broadcast usually requires multiple messages.
- Nodes usually contain a high percentage of mainframes.

COMMUNICATION

When designing a communication network, numerous issues must be addressed:

Naming and name resolution
- How do two processes locate each other in order to communicate?

Routing Strategies
- How are messages sent through the network?

Connection Strategies
- How do two processes send a sequence of messages?

Contention
- Since the network is a shared resource, how do we resolve conflicting demands for its use?

NAMING AND NAME RESOLUTION

- Naming systems in the network.
- Address messages with the process-id.
- Identify processes on remote systems by < hostname, identifier > pair.
- **Domain name service** -- specifies the naming structure of the hosts, as well as name to address resolution (internet).

ROUTING STRATEGIES

FIXED ROUTING

- A path from A to B is specified in advance and does not change unless a hardware failure disables this path.
- Since the shortest path is usually chosen, communication costs are minimized.
- Fixed routing cannot adapt to load changes.
- Ensures that messages will be delivered in the order in which they were sent.

VIRTUAL CIRCUIT

- A path from A to B is fixed for the duration of one session. Different sessions involving messages from A to B may have different paths.
- A partial remedy to adapting to load changes.
- Ensures that messages will be delivered in the order in which they were sent.

DYNAMIC ROUTING

- The path used to send a message from site A to site B is chosen only when a message is sent.
- Usually a site sends a message to another site on the link least used at that particular time.
- Adapts to load changes by avoiding routing messages on heavily used path.
- Messages may arrive out of order. This problem can be remedied by appending a sequence number to each message.

CONNECTION STRATEGIES

- Processes institute communications sessions to exchange information.
- There are a number of ways to connect pairs of processes that want to communicate over the network.

Circuit Switching	A permanent physical link is established for the duration of the communication (i.e. telephone system.)
Message Switching	A temporary link is established for the duration of one message transfer (i.e., post-office mailing system.)
Packet Switching	Messages of variable length are divided into fixed-length packets that are sent to the destination. Each packet may take a different path through the network. The packets must be reassembled into messages at they arrive.

- Circuit switching requires setup time, but incurs less overhead for shipping each message, and may waste network bandwidth.
- Message and packet switching require less setup time, but incur more overhead per message.

CONTENTION

Several sites may want to transmit information over a link simultaneously. Techniques to avoid repeated collisions include:

CSMA/CD.

- Carrier sense with multiple access (CSMA) collision detection (CD)

- A site determines whether another message is currently being transmitted over that link. If two or more sites begin transmitting at exactly the same time, then they will register a CD and will stop transmitting.

- When the system is very busy, many collisions may occur, and thus performance may be degraded.

- (CSMA/CD) is used successfully in the Ethernet system, the most common network system.

Token passing.

- A unique message type, known as a token, continuously circulates in the system (usually a ring structure).
- A site that wants to transmit information must wait until the token arrives.
- When the site completes its round of message passing, it retransmits the token.

Message slots.

- A number of fixed-length message slots continuously circulate in the system (usually a ring structure).
- Since a slot can contain only fixed-sized messages, a single logical message may have to be broken down into smaller packets, each of which is sent in a separate slot.

DESIGN STRATEGIES:

The communication network is partitioned into the following multiple layers:
<<< FIGURE 15.11 >>>

Physical layer

- Handles the mechanical and electrical details of the physical transmission of a bit stream.

Data-link layer

- Handles the frames, or fixed-length parts of packets, including any error detection and recovery that occurred in the physical layer.

Network layer

- 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.

Transport layer

- 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.

Session layer

- Implements sessions, or process-to-process communications protocols.

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 **<<< FIGURE 15.13 >>>**