

2.2 Interfacing Computers (Networking)

Network

- Nodes: Points in network that sends / receives data; end points
- Network: Interconnected system of nodes with links established via wired/wireless media
- Networking: Practice of linking two or more devices together for sharing of resources
 - Facilitate sharing of resources
 - Remote access
 - Consolidate information
- Benefits:
 - Efficiency in management of multiple machines
 - ◆ Reduce cost of maintenance / operation
 - Collaborative
 - ◆ Allows sharing of resources / data
 - ◆ No need for duplication of files / hardware
 - ◆ E.g. printer, storage, files, bandwidth
 - Convenient
 - ◆ Ease of access to resources and information
 - ◆ Allows for remote access
- Drawbacks:
 - Network failure: slow service
 - High cost due to hardware and high bandwidth required
 - Complicated set-up and maintenance
 - Expertise required
 - Security problems:
 - ◆ Controlled access
 - ◆ Intrusions
 - ◆ Virus infections
- **Ethernet**
 - A series of network devices connected together on a shared Ethernet cable
 - ◆ May or may not be connected to the Internet
 - Collision may occur when two or more devices try to send data simultaneously
 - Each device has a Media Access Control (MAC) Address
 - ◆ Physical address
 - ◆ 6 bytes

- ◆ Unique for each device
- Carrier Sense Multiple Access with Collision Detection (CSMA/CD)
 - ◆ Devices take turns to send and receive data
 - ◆ Process:
 1. Is my frame ready for transmission? Y - go to 2
 2. Is medium idle? N - wait until ready
 3. Start transmitting and monitor for collision during transmission
 4. Did collision occur? Y - go to Collision Detected Procedure
 5. Reset retransmission counter and end transmission
- Collision Detected Procedure:
 - ◆ Used to resolve a detected collision
 - ◆ Complete when retransmission is initiated or retransmission is aborted due to numerous collisions
 - ◆ Procedure:
 1. Continue transmission until minimum packet time is reached to ensure all receivers detect collision
 2. Increment retransmission counter
 3. Was max number of transmission attempt reached? Y - abort transmission
 4. Calculate and wait random backoff period based on number of collisions
 5. Re-enter main procedure at stage 1 of transmission process
- Methods to overcome collision:
 - ◆ Use switches instead of hubs: data sent to port that is meant only for device connected to that port
 - ◆ Cables:
 - ◆ Twisted pair cabling: separate cables for sending and receiving
 - ◆ Optic fibre: uses light transmission, bi-directional
- **Local Area Network (LAN)**
 - Connected to Intranet using Switches
 - Network that connects computers and devices in a limited geographical area e.g. home
 - WLAN: LAN using wireless signals
 - ◆ E.g. Bluetooth, Wi-Fi
 - Can be a part of WAN
 - Cheaper than WAN

- Able to control security, lower risk of data leaks and viruses
- Provides Intranet
 - ◆ Private network contained within an enterprise / organisation
 - ◆ Consists of many linked LANs
 - ◆ Use leased lines in WAN
 - ◆ Connections through one or more gateway computers to the Internet
 - ◆ Purpose:
 - ◆ Share information, resources, operational systems, computing services, etc. within organisation
 - ◆ Facilitate group work
 - ◆ Teleconferences
 - ◆ Tunnelling:
 - ◆ Virtual Private Network (VPN)
 - ◆ Send private data through public network
 - ◆ Uses public network with encryption/decryption and other security safeguards
 - ◆ Firewall:
 - ◆ Access to internet through firewall servers
 - ◆ Ability to screen data in both directions
 - ◆ Maintain company security:
 - ◆ Prevents viruses from entering
 - ◆ Prevents private information from being leaked
- **Wide Area Network (WAN)**
 - Connects to the Internet using Routers
 - Network that covers a large geographical area using a communications channel
 - Provides Internet
 - Devices use Internet Protocol (IP) Address
 - Combines many types of media
 - Larger volume of traffic
 - More expensive than LAN
 - Less secure due to larger exposure

Connecting Devices

- Purpose of connecting devices
 - Transferring data from one host to another
- Broadcast: receive from one and send to all
- Unicast: receive from one and send to another

- Multicast: receive from one and send to many (not all)
 - Controlling network traffic
- **Network Hub** (Physical Layer)
 - Used as network connecting device to connect all devices together
 - Basic broadcasting: replicate incoming data and forwards to all connected devices
 - Each device is responsible for determining which packets are destined for it, ignores others
 - Benefits:
 - ◆ Cheap
 - ◆ Simplicity
 - ◆ No error management: high speed
 - Drawbacks:
 - ◆ Wastes bandwidth: costs unnecessary traffic and collisions
 - ◆ Security leaks: every host/device on network gets access to data
- **Switch** (Data Link Layer)
 - Used in LAN / Intranet
 - Each device is connected to a unique port
 - MAC table: records MAC addresses of all the connected devices
 - Sends data to the addressed device (Unicast)
 - ◆ Packet sent to switch is read to determine which computer to send to
 - ◆ If switch doesn't recognise the destination MAC address, broadcast packet, each device determines if the packet is meant for it
 - Benefits:
 - ◆ Packet handling
 - ◆ Only sends packet to specified destination, better security
 - ◆ Collision management during high traffic
 - ◆ Creates connection between sender and receiver hosts
 - ◆ Improves performance and efficiency
 - ◆ Problem isolation
 - Circuit Switching
 - ◆ Two network nodes establish a dedicated communication channel through the network before nodes may

communicate

- ◆ Entire circuits are switched to route traffic to correct destination
- ◆ Benefits:
 - ◆ Circuit guarantees full bandwidth of channel, no interruption
 - ◆ Remains connected for duration of communication session
 - ◆ Highly reliable (than packet switching)
 - ◆ Ensures data gets across fully
- ◆ Drawbacks:
 - ◆ Single point of failure cause full disruption in communications
 - ◆ Expensive and inflexible due to unused capacity
- ◆ E.g. early analog telephone network
- Packet Switching
 - ◆ Transmitting data by splitting it into smaller packets
 - ◆ Groups all transmitted data into suitably sized packets (broken up)
 - ◆ Each packet consist of header and payload
 - ◆ Header: IP address, destination MAC address
 - ◆ Benefits:
 - ◆ Efficient
 - ◆ Stable: prevents single point of failure
 - ◆ Packets are short
 - ◆ Communication links between nodes are only allocated to transferring a single message for short period of time while transmitting each packet
 - ◆ Longer messages require series of packets to be sent, but do not require link to be dedicated between transmission of each packet
 - ◆ Pipelining
 - ◆ Multiple transmissions that do not cause collision can occur simultaneously
 - ◆ Drawbacks:
 - ◆ Lagging due to high traffic
 - ◆ Interference
 - ◆ Less reliable: loss of packets resulting in data not fully transmitted

- **Router** (Network Layer)

- Used for WAN / Internet

- Connect networks to one another using IP address
- Receives packet, router reads destination IP, forwards to destination device
- Can assign IP addresses using DHCP
- Can be used as gateway

- **Gateway**

- System that joins two or more networks with different protocols
- Connects Ethernet to Internet
- Implementation:
 - ◆ Completely in software
 - ◆ Completely in hardware
 - ◆ Combination of software and hardware
- Operates on every level of OSI model
- Firewalls, VPNs, etc can be integrated into gateways
- E.g. broadband router in households

Network Architecture

- **Client-Server**

- Specific workstations/terminals that serve the requests of other systems
- One or more computers act as server
- Purpose:
 - ◆ Provides service to other systems
 - ◆ Dedicated to one server task
- Client:
 - ◆ Request services from the specific servers
- E.g. File server, Print server, Mail server, Proxy server, Domain Name server, Network Time Protocol server
- Benefits:
 - ◆ Centralisation
 - ◆ Central servers
 - ◆ Help in administrating set-up
 - ◆ Access rights and resource allocations by server
 - ◆ Proper file management
 - ◆ All files stored in servers
 - ◆ Easy management
 - ◆ Back-up and recovery
 - ◆ Security
 - ◆ Rules of security defined when setting-up server

- Drawbacks:
 - ◆ Congestion
 - ◆ High volume of traffic overloads servers
 - ◆ Servers break down
 - ◆ Service unavailable
 - ◆ Service is down when servers fail
 - ◆ Maintenance
 - ◆ Professionals are required to maintain the set-up
- **Proxy Server**
 - ◆ Obscure client IP, provides anonymity
 - ◆ Bypass IP address blocking
 - ◆ Firewall: filters request to control incoming and outgoing data
 - ◆ Block malicious traffic
 - ◆ Log activities
 - ◆ Improve performance:
 - ◆ Browser send all HTTP request to cache.
 - ◆ If data in cache, proxy returns cache to client; else proxy fetch data from internet and returns data to client, and stores it in proxy cache
- **Domain Name Server**
 - ◆ Contain database of domain names and their IP addresses
 - ◆ Hierarchy of databases
 - ◆ Device sends domain name to DNS
 - ◆ Process:
 - ◆ DNS checks its cache if it contains requested domain's IP address:
 - ◆ If present, returns back to client
 - ◆ If not present, DNS queries Internet Service Provider (ISP)
 - ◆ If ISP does not contain requested domain:
 - ◆ Direct query to Root server
 - ◆ If Root server does not contain requested domain:
 - ◆ Direct query to Top Level Domain (TLD) server
- **Mail Server**
 - ◆ Outgoing:
 - ◆ Simple Mail Transfer Protocol (SMTP)
 - ◆ Incoming:

- ◆ Post Office Protocol 3 (POP3)
 - ◆ Stores emails on client device
 - ◆ Delete email after forwarding to client
- ◆ Internet Message Access Protocol (IMAP)
 - ◆ Stores copies of emails online
- ◆ Process of sending mail:
 - ◆ Client sends mail to SMTP server of his own domain
 - ◆ SMTP server reads recipient address to determine domain
 - ◆ If sender and receiver have the same domain, SMTP forwards the email to the domain's POP3 or IMAP servers
 - ◆ Sender SMTP server looks for recipient SMTP server and forwards mail to it
 - ◆ Recipient SMTP server forwards email to POP3 for local storage or IMAP for online storage
 - ◆ Email ready for download when recipient client comes online
- **Dynamic Host Configuration Protocol (DHCP)**
 - ◆ Every device has a unique unicast IP address to access network
 - ◆ Centralised and automated TCP/IP configuration
 - ◆ Assigns IP addresses to connected devices automatically
 - ◆ Allocation of IP address (DORA):
 1. DISCOVER: Client broadcasts DHCPDISCOVER packet in subnet
 2. OFFER: Server responds with DHCPOFFER packet containing potential IP addresses for client
 3. REQUEST: Client responds with DHCPREQUEST packet containing server identifier, if multiple offer packets are received, client chooses the fastest response
 4. ACK: Server replies with DHCPACK packet to acknowledge client on requested IP address, stores IP address into database
 - ◆ DHCP servers maintain a pool of IP addresses and leases an address to a client when it connects to the network
 - ◆ Addresses that are no longer in use are automatically returned to the pool
 - ◆ Benefits:
 - ◆ Reliable

- ◆ Minimises errors caused by manual configurations
 - ◆ Reduce network administration
 - ◆ Efficient
- **Peer-to-Peer (P2P)**
 - Each computer (peer) has equal responsibilities and capabilities.
 - No central server
 - All computers able to share resources without going through a server computer
 - E.g. bitTorrent, Napster
 - Benefits:
 - ◆ More resilient in case of failures and traffic bottlenecks
 - ◆ Lack of central server
 - ◆ Cheaper costs
 - ◆ No maintenance needed
 - ◆ More available resources
 - Drawbacks:
 - ◆ No control over shared data
 - ◆ Copyright infringements, piracy
 - ◆ Poor security
 - ◆ Virus
 - ◆ Illegal access to computer by others
 - ◆ User computer can be slowed down when accessed by others

Transmission

- **Rate of Transmission**
 - Measure of the amount of data that can be transmitted through a connection over a given amount of time.
 - Every machine is connected by cable or other types of connection
 - Measure of bandwidth
 - ◆ Unit: bits per second (bps)
 - ◆ aka Baud Rate / Bit rate
- **Directions of Transmission**
 - Simplex:
 - ◆ Data transmission in one direction
 - ◆ E.g. keyboard: keyboard transmits user input to CPU, but CPU does not need to reply to keyboard
 - Half-Duplex:

- ◆ Transmission in both directions, but only one direction will be allowed through at a time
- ◆ Type of parallel interface
- ◆ Consists of 8 lanes
- ◆ E.g. printers, walkie-talkie
- Full-Duplex:
 - ◆ Transmission in two ways simultaneously
 - ◆ E.g.:
 - ◆ Telephones: allow both people to hear each other at the same time
 - ◆ Computers: connected via Ethernet cable can send and receive data at the same time
 - ◆ I/O standards: USB / Thunderbolt

● Synchronisation of Transmission

- Coordinating sending and receiving within the network
- Serial transmission: transmission by single bits
- Asynchronous transmission:
 - ◆ Sends only one character at a time
 - ◆ Character either an alphabet, number, or control character
 - ◆ Bit synchronisation between two devices:
 - ◆ Use of start bit and end bit
 - ◆ Indicated beginning and end of data
 - ◆ Idle time between transmissions of different data types
 - ◆ Responsibility of sender to separate bit stream into bytes of characters
 - ◆ Sender and receiver are not to be synchronised
 - ◆ Receiver has to synchronise with incoming bit stream when receiving
 - ◆ Benefit:
 - ◆ Low cost
 - ◆ Drawback: Slow transmission speed
- Synchronous transmission:
 - ◆ Bit stream is combined into longer frames that may contain multiple bytes
 - ◆ No gaps between various bytes in data stream
 - ◆ Bit synchronisation established between sender and receiver by timing transmission of each bit
 - ◆ Sender and receiver operate at same frequency in order for receiver to receive data error-free
 - ◆ Responsibility of receiver to separate bit stream into bytes

so as to construct original information

- ◆ Benefit:
 - ◆ Fast transmission speed
- ◆ Drawback:
 - ◆ High costs

Cyclic Redundancy Check

- Purpose: Check that information is entered correctly / transmitted without corruption
- **Parity check**
 - Uses parity bit
 - Using bit patterns
 - ◆ Even parity check: Parity bit added to ensure even number of "1"s
 - ◆ Odd parity check: Parity bit added to ensure odd number of "1"s
 - Used for small blocks of data
 - Simple to check
 - Detects error when bit parity is wrong
 - Unable to detect error when two bits are altered
- **Check digit**
 - Attach weights to the digits
 - Sum the product of each weight to the corresponding digit of the code
 - Divide the sum using the modulo to find remainder
 - Check digit is the difference between modulo and remainder
 - Check digit added to the back of the code
 - E.g. Modulo 11
 - ◆ Weights: 7, 6, 5, 4, 3, 2
 - ◆ Code: 508795
 - ◆ Modulo: 11
 - ◆ Weighted sum: $7 \times 5 + 6 \times 0 + 5 \times 8 + 4 \times 7 + 3 \times 9 + 2 \times 5 = 140$
 - ◆ Remainder: $140 / 11 = 12 \text{ R } <8>$
 - ◆ Check digit: $11 - 8 = 3$
 - ◆ Code: 5087953
 - For checking:
 - ◆ Find weighted sum of the multiplication of code and weight, check digit has
 - ◆ Divide by modulo
 - ◆ Weighted sum should be exactly divisible by modulo (no

remainder)

- ◆ Check digit has weightage of 1
- ◆ E.g. Modulo 11
 - ◆ Code = 5087953
 - ◆ Weighted sum = $7 \times 5 + 6 \times 0 + 5 \times 8 + 4 \times 7 + 3 \times 9 + 2 \times 5 + 1 \times 3 = 143$
 - ◆ Remainder = $143 \bmod 11 = 0$
 - ◆ Thus valid code
- Used for small blocks of data
- **MD5 Checksum**
 - Hash function to check if file is legit and untampered with
 - 128 bits string
 - If original and checked MD5 checksums match, the file is legit
 - If MD5 strings do not match, there's a possibility that the file has been altered
- Most common approach after error detection: Retransmit

Cloud Computing

- **Traditional server**
 - Organisations own their own servers
 - Drawbacks:
 - ◆ High initial setup cost
 - ◆ Require maintenance
 - ◆ High level of administrative work
- **Cloud Computing**
 - Network computing approach whereby applications run on a server / group of servers owned by a service provider
 - Technologies that provide software, data access, and storage services
 - Not owned by user
 - Virtualisation:
 - ◆ Technique of running OS and software within another OS / software
 - ◆ Simulates real hardware
 - ◆ Allows multiple virtual machines to run on the same set of hardware
 - ◆ Foundation of cloud computing
 - Benefits:
 - ◆ Utility based usage

- ◆ Pay-as-you-use approach
- ◆ Use when you want to
- ◆ Does not require user to configure or understand how the system works
- ◆ Flexibility:
 - ◆ Scalability and Elastic capabilities: Support fluctuating workloads, users can request for more services
 - ◆ Control choices: Services offer different levels of control as they require
 - ◆ Security: Imbedded virtual private clouds, encryption, etc.
- ◆ Efficiency:
 - ◆ Ease of access: Cloud based applications and data
 - ◆ Data security: Hardware failures do not result in data loss due to cloud back-ups
 - ◆ Pay structure: Utility based usage, pay for what you use
- ◆ Strategic Value:
 - ◆ Collaboration: Worldwide access allows users to collaborate from various locations
 - ◆ Competitive edge: Ability to devote less resources to managing infrastructure
 - ◆ Economy: decrease in total cost per unit as size of system grows
- Drawbacks:
 - ◆ Lack of control:
 - ◆ No ownership and control over hardware and infrastructure
 - ◆ Question of ownership: if data is owned by user or service provider
 - ◆ Freedom of usage limited by producer e.g. downed services
 - ◆ Security:
 - ◆ Data that used to reside locally is now stored and resided on the cloud
 - ◆ Data is now openly accessible as it is put on the Internet
 - ◆ High cost:
 - ◆ Cloud computing may be efficient for large companies
 - ◆ Small companies may lose money due to low utilisation

- Cloud deployment models:
 - ◆ Private cloud
 - ◆ Public cloud e.g. Google Suite
- Cloud service models:
 - ◆ Software as a Service (SaaS) (Top Tier)
 - ◆ Allows users to run prebuilt online applications
 - ◆ Hardware and software provided and managed by service provider
 - ◆ Can't be customised more than provider allows
 - ◆ No administrative work needed
 - ◆ E.g. iCloud, Google Suite
 - ◆ Platform as a Service (PaaS) (Mid Tier)
 - ◆ Allows users to create, edit, maintain their own cloud applications using supplier-specific tools and languages
 - ◆ Hardware is provided and managed by service provider
 - ◆ Operating system and applications managed by user
 - ◆ Low administrative work: applications, data
 - ◆ E.g. Google App Engine, Pivotal CF
 - ◆ Infrastructure as a Service (IaaS) (Base Tier)
 - ◆ Allows users to run any application they please on cloud hardware of their choice
 - ◆ Grants users full access to hardware
 - ◆ Virtual machines
 - ◆ Hardware is provided and managed by service provider
 - ◆ High administrative work: applications, data, OS
 - ◆ E.g. Amazon Web Services; Google; Windows Azure

