The ISO OSI Reference Model

Overview: OSI services

Physical layer

Data link layer

Network layer

Transport layer

Session layer

Presentation layer

Application layer

Overview

- Formal framework for computer-to-computer communications
- Standardized by ISO
- Layered organization:
  - services: what the layers do
  - interfaces: how to access (use) the services
  - protocols: private peer-to-peer messages
- References:
  - Tanenbaum: Ch 1.3, 1.4, 1.6, 2.2, 2.4, 2.5, 3.1, 3.6, 5.1, 5.2, 5.4, 6.1, 6.2
  - Kurose & Ross: Ch 5.1, 5.8, 5.9.2, 5.9, 5.10, 4.1, 4.2, 4.6, 5.6, 8.5, 3.1.1, 2.1.1, 2.1.2

The big picture
(Tanenbaum, pg 29)

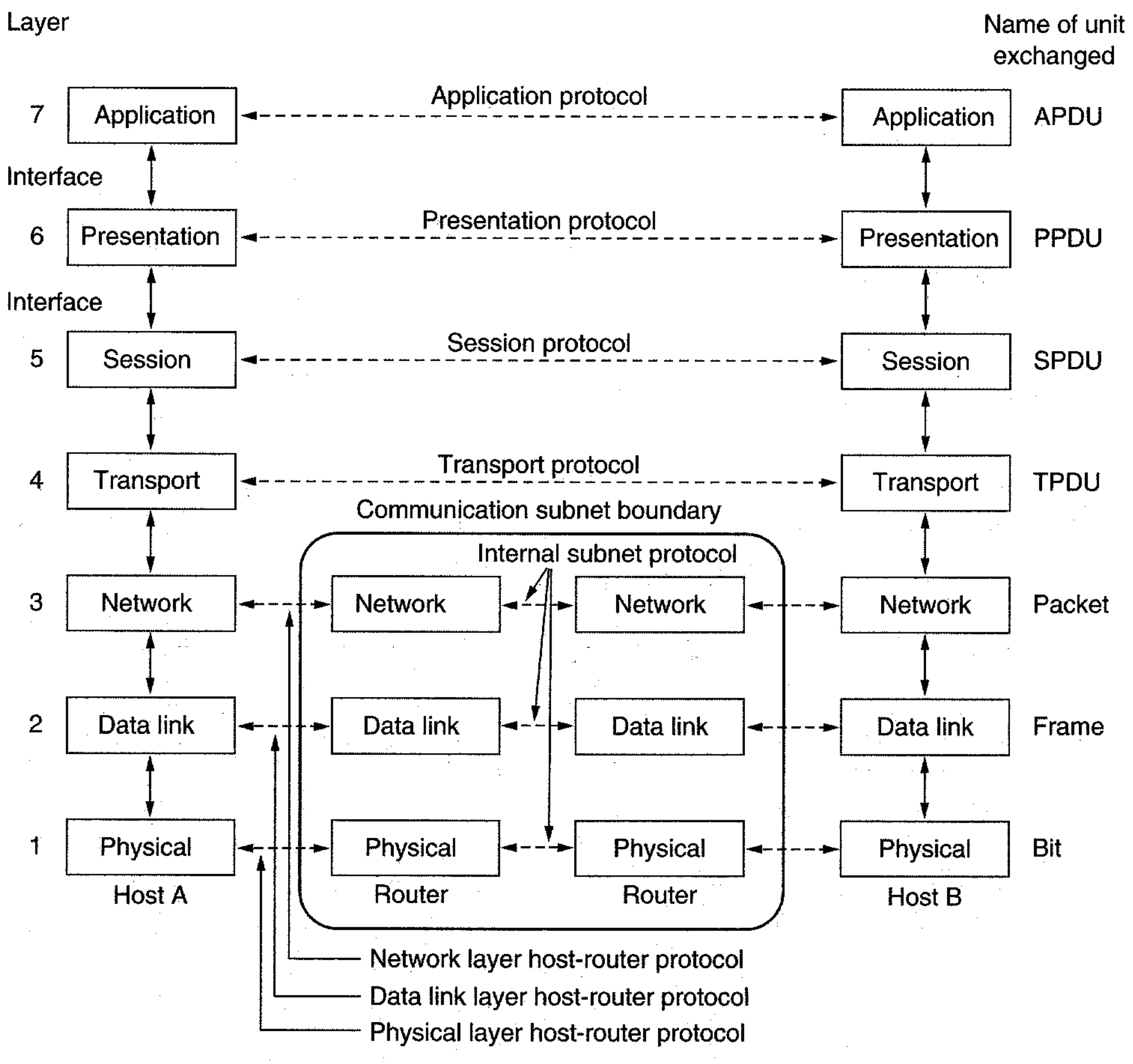


Fig. 1-16. The OSI reference model.

Services

- A service:
  - is a set of functions provided by a layer to its above layer
  - defines the operations a layer can perform on behalf of its users
  - service definition does not say how these operations are implemented
- A protocol:
  - is a set of rules governing the format and meaning of the frames, packets or messages exchanged by the peer entities within a layer
  - implements a service
- So:
  - any protocol that implements a given service can be used where required
    - network products & systems can change protocols at will provided the service definitions do not change

continued...

- Functional aspects of OSI services:
  - connection-oriented (CO) vs. connectionless (CL)
    - two varieties of CO services: message stream and byte stream
  - reliable vs. unreliable
  - a virtual-circuit service is an example of a connection-oriented service
  - a datagram service is an example of a a connectionless service
- A layer specification can define one or many services, e.g.
  - virtual circuits of varying characteristics
  - “datagram service” for unacknowledged vs. “acknowledged datagram service” (Tanenbaum)
    - “request–reply”: special version of acknowledged datagram

Commentary

- The telephone–telecommunications sector has had a substantial influence on computer networking design
- World telecommunications standards are governed by ITU-T (formerly CCITT), a UN agency
- ITU-T is composed of members of national PTTs, private operators (ATT, Bell, BritishTel, etc.), associated organizations (notably ISO)
- OSI was created by ISO, but with significant influence of telephony:
  - connection-oriented services
  - service definitions (e.g. “indication”)
  - telephony and computing worlds can’t even agree on the meaning of “open” and “closed”!

Aside: service primitives

- OSI service primitives are basic verbs used to define services:
  - request: initiate an action
  - indication: be informed about an action
  - response: respond to an action
  - confirm: acknowledgement of a previous action
- E.g. the definition of a CONNECT service (a service to create a connection):
  - host Y wants to connect to host X
    - Y.CONNECT.request(X): node Y request a connection to node X
    - X.CONNECT.indication(Y): network advises X of incoming connection from Y
    - X.CONNECT.response: X accepts or rejects
    - Y.CONNECT.confirm: network advises Y of success or failure

… service primitives, 2

- This is an awful abstraction for computing
  - interrupt-driven (asynchronous)
  - telephony’s ideal situation
    - telephones rings, interrupts you, and you service the interrupt

Physical layer

- Define the physical characteristics of the transmission medium
  - define the hardware-software interface
- Example: RS232-C standard for computer-to-modem communications
  - four specifications: mechanical, electrical, functional and procedural
  - mechanical: connector shape, number pins
  - electrical: voltages, cable lengths
  - functional: pin assignments and meaning (CTS, RTS, DTR, etc)
  - procedural: describes how to do operations
    - to send to modem, computer “raises” RTS
    - modem raises CD when call received

Physical layer: media

- Other physical layer media:
  - magnetic media (tapes): really high bandwidth, really long transmission delay
  - twisted-pair: common, really cheap, Mbps for 10's Km, error-prone
  - baseband coaxial: digital signalling, fast–Gbps for 1km, cheap
  - broadband coaxial: analog signalling, 50-100 6Mhz channels, ~100km, unidirectional, cheap
  - fiber-optic: 50,000 Gbps, electro-optic conversion, expensive
  - PSTN: public switched telephone network
  - plus: line-of-sight microwave & infrared; broadcast & point-to-point radio, cellular, satellite; etc.

Physical layer: media-2

- For LANs, media is usually owned
  - some variation of twisted-pair copper is the overwhelmingly common choice today
  - fiber is sometimes used for backbone, NOC and mainframe networks
- For WANs & MANs, generally leased:
  - "dark" or "unloaded" leasing of bare physical medium
  - purchasing of network services
    - not strictly just physical layer, but can be considered as "value added" medium
  - the simplest is the PSTN (aka POTS: Plain Old Telephone System)
    - switched point-to-point copper twisted pair for analog signaling
    - data rates up to 56 Kbits/sec
  - for wider WAN deployments, there is an alphabet soup of available technologies & services: SONET/SDH, POS, X.25, frame relay, T1, T3, OC-*x*, DS-*x*, SMDS, xDSL, B-ISDN/ATM, HFC, …

Alphabet soup

- Two categories of media: switched-network services and point-to-point
- Point-to-point technologies:
  - T1, T3: TDM packet technology for voice & data
    - T1: 1.54 Mbits/sec; T3: 44.736 Mbits/sec
    - data interface is called a DSU/CSU (data service unit/channel service unit) which acts as a converter between computer & telephony physical layers
    - terms DS-1 & DS-3 refer to data rates, T1 & T3 refer to interface specifications
    - one DS-1 channel carries 24 simultaneous voice channels
    - T2/DS-2 and T4/DS-4 also exist
  - these numbers are North American, European standards are slightly different
    - e.g. ITU-T Level 1 is 30 voice channels for a total of 2.048 Mbits/sec
    - T1 etc are called E1 etc in Europe

…alphabet soup, 2

- Point-to-point technologies:
  - SONET/SDH (Synchronous Optical Network/Synchronous Digital Hierarchy): high-speed (51.84 to 2,488 Mbits/sec)
    - basic SONET is OC-1 (Optical Carrier 1)
    - OC-*n* refers to multiples of OC-1, e.g. OC-3 is 155 Mbits/sec
      - OC-48: 2.5Gbps; OC-192: 10Gbps
    - STS-*n* is the electrical equivalent of OC-*n* (Synchronous Transport Signal)
    - European standards are the same speeds, but different names (STM-*n*)
  - xDSL (Digital Subscriber Lines)
    - various forms: A for asymmetric, S for symmetric, H for high-rate
    - uses FDM over standard copper pairs, required special-purpose modems
    - varying rates; depends on version, direction, distance; between 32 Kpbs and 6.4 Mbps

…alphabet soup, 3

- Point-to-point technologies:
  - HFC (Hybrid Fiber Coaxial) provided by cableco
    - combination of fiber OC circuits to neighbourhood distribution centres, then FDM and TDM over existing cable plant to modems
      - FDM to separate TV, upstream data and downstream data
      - TDM within each data-stream
    - asymmetric data rates, varying between 1.5 Mbits/sec down and 6Mbits/sec up
    - not distance sensitive
    - may require significant infrastructure upgrades by cableco

…alphabet soup, 4

- Switched-network technologies:
  - X.25
    - first commercial packet-switched network: Datapac in Canada
    - data rates up to 56 Kbits/sec
    - obsolete now
  - ISDN (Integrated Services Digital Network)
    - digital version of PSTN
    - data rates depend on provider product definition – typically 128 Kbits/sec
    - obsolete (or rapidly becoming so)
  - Frame relay
    - newer version of X.25
    - data rates from DS-1 to DS-3 (1.5 to 44 Mbits/sec)
  - SMDS (Switched Multi-megabit Data Service)
    - commercial repackaging of frame relay
  - [ATM discussion deferred]

How to choose

- On what basis can a rational decision be made about selecting appropriate physical media or services for WAN deployments
  - e.g. how do I choose between T1 and frame relay?
- In general, the point-to-point technologies are *distance-sensitive:* the cost increases as the distance increases
- Switched-network services are not distance-sensitive, but often have volume or capacity fee structures
- Cost analysis will require
  - required data rates
  - required/expected volume
  - distances
  - telco and cableco tariff fine print

Data link layer

- Use physical layer to transmit/receive
  - “ok, I have a physical medium, what do I do with it?
- Provide error-free transmission to network layer
  - differing service classes
  - perform error-checking and request retransmissions if needed
- Divide data-stream into frames
  - create and recognize frame boundaries
- Control flow between senders and receivers
- Other issues:
  - efficient use of medium
  - controlling access to physical layer in broadcast networks: *Medium Access Control* (MAC) layer

Data link layer: services

- Unacknowledged connectionless
  - no ACKs
  - no connections
  - lost frames are really lost
  - useful in real-time applications
- Acknowledged connectionless
  - no connections
  - each frame is ACKed
- Connection-oriented
  - reliable packet-stream for network layer
  - connection (virtual circuit) is set up
  - frames are transmitted
  - connection is released

Data link layer: framing

- Datalink layer receives packets from network layer and transmits contents as bitstream via physical layer
- Sender breaks up stream into frames for calculation of FCS (or similar)
- Receiver must reconstruct frames from physical-layer bitstream
- *Framing* is non-trivial: how does receiver recognize frame boundaries?
  - transmission errors can destroy beginning, middle, end of frame
  - may not even know how many bits lost
  - timing is not feasible
- Several standard methods:
  - start/end characters, character stuffing
  - start/end bit patterns, bit stuffing
  - physical-layer encoding

Data link layer: framing-2

- Start/end characters with character stuffing
  - precede each frame with ASCII characters DLE STX
    - DLE == Data Link Escape
    - STX == Start TeXt
  - follow each frame with DLE ETX
    - ETX == End TeXt
  - if an error occurs and frames become unsynchronized, skip until next DLE STX (or DLE ETX)
- What if DLE STX or DLE ETX occur in the “data”?
  - ambiguous
  - solution: insert an extra DLE beside the “accidental” DLE
  - technique is called character stuffing
- Significant disadvantage: dependent on 8-bit characters and ASCII

Data link layer: framing-3

- Start/end bit patterns with bit stuffing
  - general version of character stuffing independent of character set
  - use special bit pattern to mark frame beginning and end: 01111110 (six 1s)
  - whenever frame contains a sequence of five 1s, insert (stuff) a single 0 bit
  - thus, six 1s can occur only for frame boundaries
  - receiver scans bitstream; if it sees five 1s followed by a 0, removes (destuffs) the 0
  - any occurrence of six 1s must (should) be surrounded by 0s and implied a frame boundary
- Physical-layer encoding
  - rely on physical-layer protocols to use redundancy: e.g. transmit each data bit as two physical bits

Data link layer: example protocols

- HDLC
  - high-level data link control, from ISO
  - based on IBM's SDLC (synchronous data link control) layer in SNA
  - various permutations
    - ANSI: ADCCP (advanced data communications control procedures)
    - CCITT: LAP (link access protocol), LAP-B; middle layer in X.25
  - all varieties more or less the same:
    - bit-stuffed
    - sliding-window
    - CRC-CCITT checksums
  - three frame types each with distinct header information (e.g. sequence numbers, piggyback ACK)
  - lots more details (see Tanenbaum)

Data link layer: example protocols, 2

- BSC
  - binary synchronous communications
  - first general-purpose multiple-station data link protocol
  - very popular, but now obsolete
- LLC
  - logical link control
  - part of the 802 family for LANs
  - based on HDLC
- LAP-D
  - ISDN version of LAP
- Internet (TCP/IP) data link protocols:
  - for use in WANs, connecting LANs together:
    - SLIP: serial line IP
    - PPP: point-to-point protocol
  - within LANs, not defined, uses IEEE 802 family of LAN broadcasting

Data link layer: example protocols, 3

- ATM (aka *cell relay*)
  - Asynchronous Transfer Mode
  - telephony world's attempt to define general-purpose protocol family to handle voice, data and video
  - ATM's core concept is dividing the datastream into small, fixed-size (53 byte) frames called *cells*
    - ATM provides multiple simultaneous datastream transmission services with defined performance characteristics:
      - CBR: constant bit rate
      - VBR: variable bit rate
      - ABR: available bit rate
      - UBR: unspecified bit rate
      - (Kurose & Ross Table 4.1)
    - ATM is a virtual-circuit network, called *virtual channels*

Data link layer: example protocols, 4

- ATM
  - ATM actually refers to three distinct layers that ~cover the range of the first four OSI layers (physical to transport)
    - ATM physical layer: handles interface to physical medium
      - designed to be independent of any particular physical medium, but telco OC-n are common, OC-3 (155 Mbits/sec is the usual starting point)
    - ATM layer: handles cell construction and management
    - ATM adaptation layer (AAL): a family of layer definitions supporting ATM service classes (CBR, VBR …)
      - other AAL definitions exist for special-purposes like IP over ATM
  - ATM had been intended to be desktop end-to-end, but is now relegated to backbone uses

Network layer

- Recall description of WAN: collection of interconnected LANS and hosts
  - subnet is the collection of network routers and communication links
- The primary job of the network layer is to move data from start to finish, by “nagivating” around the subnet
  - network layer must be aware of subnet topology: routing and internetworking
  - provide subnet-independent services to the transport layer
  - provide a uniform naming/addressing scheme for internetworking
- Since routing is handled by the network layer, so is congestion control

Network layer: services

- Services are independent of the subnet:
  - transport layer gives a TPDU to the network layer and says “deliver it”
  - network layer must decide how to get it there
  - if more that one network is involved, called *internetworking*
- Two data-transmission services:
  - reliable connection-oriented
  - unreliable connectionless
  - other two (unreliable connection, reliable connectionless) are possible but generally not offered
  - this is a complex, contentious issue

Network layer: connectionless vs. connection-oriented

- More political than technical
- Telephone people:
  - connections are good
  - connections are sophisticated – they have quality of service parameters, flow control, sequence guarantees, bidirectional
- Computer networks people:
  - network layers inevitably fail, so transport must do error and flow control themselves
  - since transport is going to do the work, network need not, so unreliable connectionless is fine
- Both arguments are valid: decision is based on where (which layer) to put the complexity
  - this is critical, since the network-transport interface is special

Network layer:
network–transport interface

- Network layer is the highest layer that knows about subnets and internetworking
- Transport layer is true host-to-host
- This layer boundary is often the boundary between network service providers and clients, e.g.
  - Frame relay is a network layer service
  - ATM has a network layer
  - IP is a network layer service
- Because this layer represents service provider–service purchaser contractual obligations and expectations, it must be defined very clearly
  - the carrier controls up to and including the network layer
  - the host controls from the transport layer up

Network layer: organization

- Consider network layer peer protocols
  - implementation models:
    - virtual circuit
    - datagram
  - choice of model effectively defines the subnet model, since the collection of network peers *is* the subnet
- So: virtual-circuit subnet or datagram subnet
- Virtual circuit subnet:
  - uses reliable connection-oriented data link services
  - uses same route for every packet in the circuit (determined once when connection established)
- Datagram subnet:
  - uses unreliable connectionless data link services
  - routes each packet independently

Network layer: comparison of virtual circuit and datagram subnets

Issue	Datagram subnet	VC subnet
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Subnet does not hold state information	Each VC requires subnet table space
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow this route
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Congestion control	Difficult	Easy if enough buffers can be allocated in advance for each VC

Fig. 5-2. Comparison of datagram and virtual circuit subnets.

- (Tanenbaum pg 344)
- Many of the differences attributed to "paying the price" at circuit setup vs. spreading the cost over every datagram

Network layer: routing

- Assumption: sending data through a subnet will require many "hops" (visits to routers)
- Routing: choosing the sequence of routers to send data from start to finish
  - goal: maximize subnet throughput
  - assumes router has more than one outgoing path: if not, easy decision
  - routing not needed inside a broadcast networks (but likely between)
- Many styles and algorithms exist:
  - global routing / link-state
    - entire network topology is known *a priori*
  - decentralized distance-vector / least-cost
    - entire network topology not known
  - others:
    - hot-potato, shortest path, least-loaded, flow-based, flooding

Network layer: routing-2

- Some common principles:
  - correct: obviously!
  - simple: if possible
  - robust: able to handle changing subnet topology (e.g. as routers crash and restart)
  - stable: opposite to robust; don’t want to over-react to subnet changes
  - fair: ensure all host are served equally
  - optimality: meet goals of throughput and packet travel
- What do we optimize?
  - number of hops
  - apparent delay times
  - link speed or throughput capacity

Network layer: routing-3

- Two major types of routing algorithms:
  - non-adaptive: (also static routing) routes are determined offline, in advance, and downloaded into routers
  - adaptive: adjusts routes as subnet information becomes available
- Non-adaptive:
  - fast: no computation required
  - inflexible: acceptable in some circumstances, unacceptable in others
- Adaptive:
  - acquires information from other routers (implies a routing-information protocol)
  - adjust routing tables accordingly
- In most practical implementations, routers exchange information, and thus formal *routing protocols* are required to facilitate these exchanges

Network layer: autonomous systems

- A global network of inter-connected networks (each with many routers) is a big network
  - global routing (knowing entire network topology) would be intractable, decentralized too slow
- Solution:
  - partition the world into a collection of autonomous systems (AS)
    - all routers in an AS use the same policies and algorithms and know about each other
      - each AS will be responsible for all routing within itself
      - AS decides itself how to do this
    - called *intra-AS* routing
  - one (or more) routers in each AS handle *inter-AS* routing to other ASs
    - routers that handle inter-AS routing are called *gateway routers*

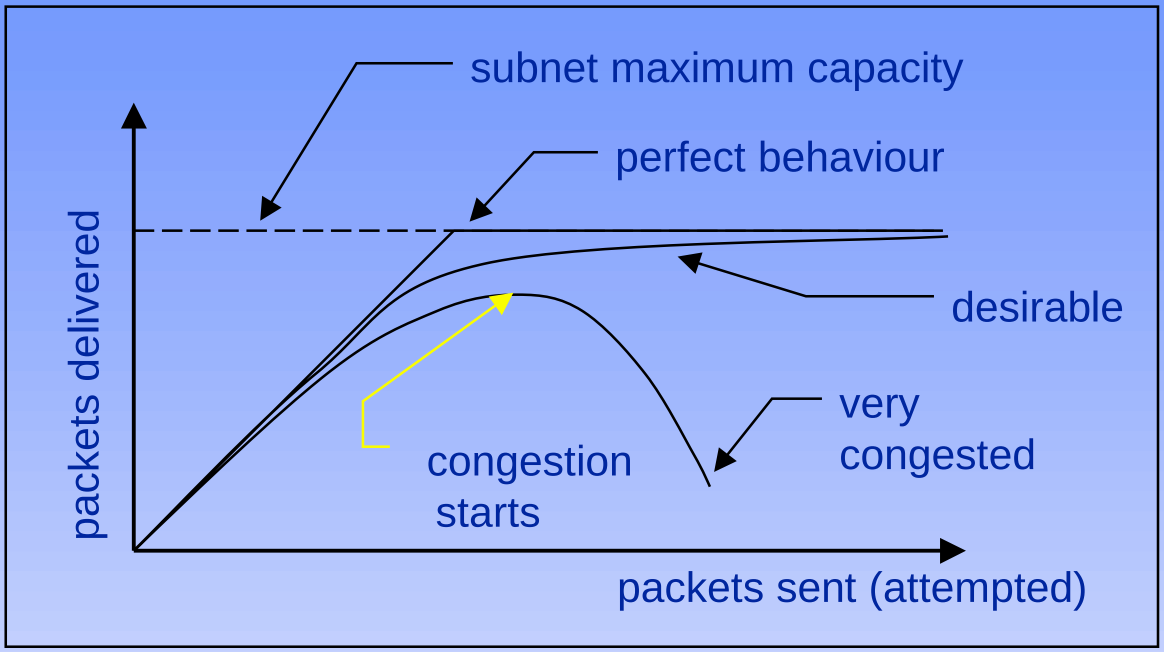
Network layer: autonomous systems, 2

- Inter-AS and intra-AS routing follow the same general principles, but vary significantly in details
  - intra-AS can assume the same link and network layers, so addresses and PDU definitions will be equivalent
  - inter-AS routing may have to handle significant differences
- How many ASs are there in the Internet?
  - North America:
    - June 1990: ~700
    - January 2001: ~20,000
      - AS 1 belongs to BBN
      - U Waterloo is AS 12093
      - ref: whois.arin.net
  - Europe, Asia-Pacific managed separately

Network layer: congestion

- Congestion: too many packets in a subnet, because
  - there are too many hosts trying to send
  - a router is too popular and becomes saturated (overwhelmed)
  - a router is too slow, or has insufficient memory to handle output queues
  - a router is too fast for an output line, and a huge queue builds
- Congestion often builds on itself
  - as congestion begins, packets are lost or delayed
  - this causes more retransmits, which makes congestion worse, so more packets are lost or delayed…
  - adaptive routing may confuse the issue, as congestion is noted

Network layer: congestion-2



- Congestion control vs. flow control:
  - congestion is global, whole subnet
  - flow control is single pair
- Congestion control principles:
  - preventative: stop congestion before it happens (or before it gets too bad)
  - reactive: determine when congestion has occurred, and fix it
  - preventative applicable for datagram services, reactive for virtual circuit

Network layer: internetworking

- The network layer is responsible for moving packets from one network to another
  - networks may use different:
    - character encodings
    - addressing schemes
    - service types (connectionless vs. connection-oriented)
    - packet sizes
    - physical-layer characteristics
  - network layer is responsible for figuring it all out
- Connections between networks usually handled by dedicated devices called:
  - repeater or hub
  - bridge or switch
  - router or multiprotocol router
  - application gateway

Network layer: internetworking-2

- There is no universal agreement on the use of these terms:
  - commercial devices combine functionality, especially bridge, switch and router
- One way to distinguish is to define according to layer:
  - physical layer is a repeater or hub
    - e.g. ethernet repeater reads & writes bits with absolutely no idea what they mean (like a “digital amplifier”)
  - data link layer is a bridge
    - store-and-forward operation
    - receive frames and re-send on another link (inspects layer-2 frame address);
    - transparent (“plug&play”) operation
    - a *switch* is analogous to a bridge, differs in commercial features

Network layer: internetworking-3

- Internetworking devices
  - network layer is a router:
    - similar to bridge: store and forward
    - Kurose & Ross focus on networks and internetworking in which TCP/IP is the only protocol family, which simplifies their definition of a router
    - Tanenbaum has a more general definition that allows for connection between differing networks
      - called “multiprotocol routers”
      - may need to be aware of packet contents
      - handles differing address schemes, packet sizes, service classes (e.g. VC numbers)

Network layer: internetworking-4

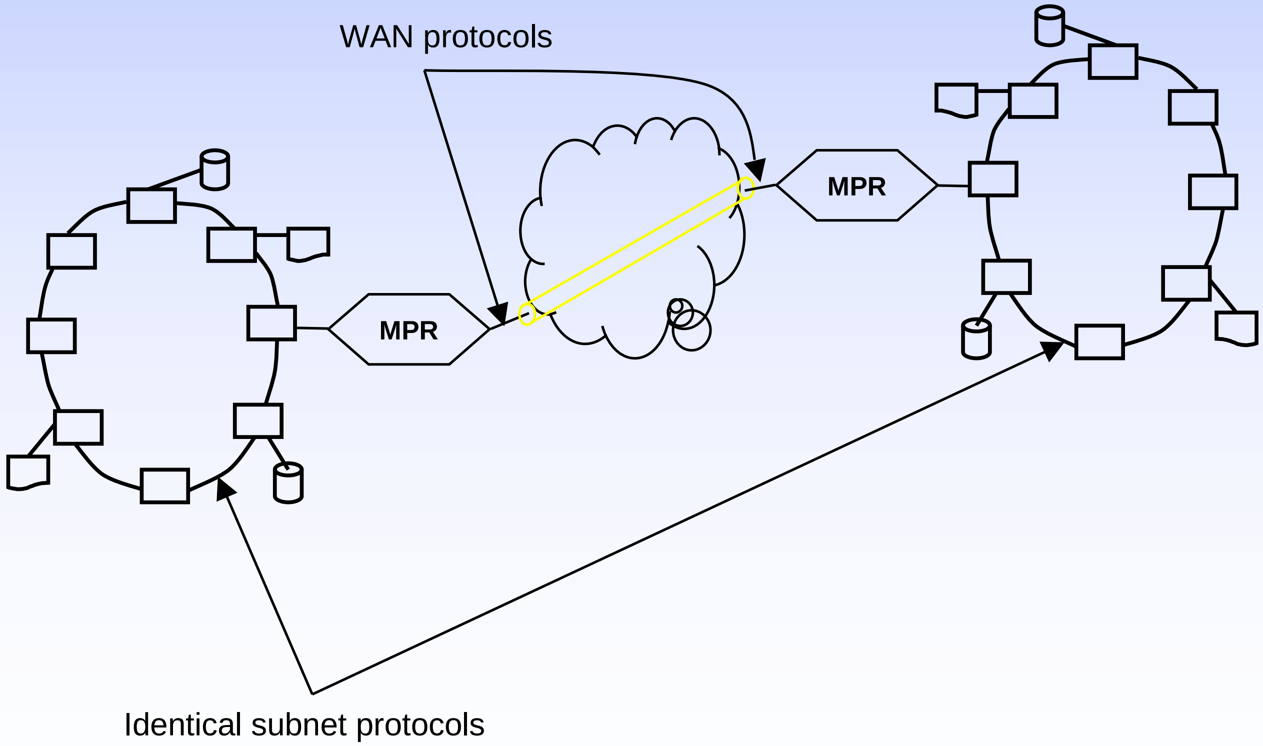
- Internetworking devices
  - above network layer is a gateway:
    - provide application services between networks
    - provide logic for converting between applications of similar purpose but (completely) different implementation
      - Lotus Notes to SMTP
    - application gateways can suffer from “least common denominator” problem
      - what to do if features and services are not defined or incompatible?

Network layer: internetworking-5

- Other inter-AS networking concepts (ref. Tanenbaum Ch 5.4.2-5.4.4, 5.4.7):
  - concatenated virtual circuits
  - connectionless internetworking
  - tunnelling
  - firewalls
- Concatenated virtual circuits:
  - build a connection from source to destination from sequence of virtual circuits in dissimilar networks
  - multiprotocol router must remember the circuits (build tables), encapsulate packets accordingly
  - achievable if networks are at least somewhat similar, difficult if not
- Connectionless internetworking
  - difficult to achieve in the general case
  - PDU, address translation not feasible in the general case

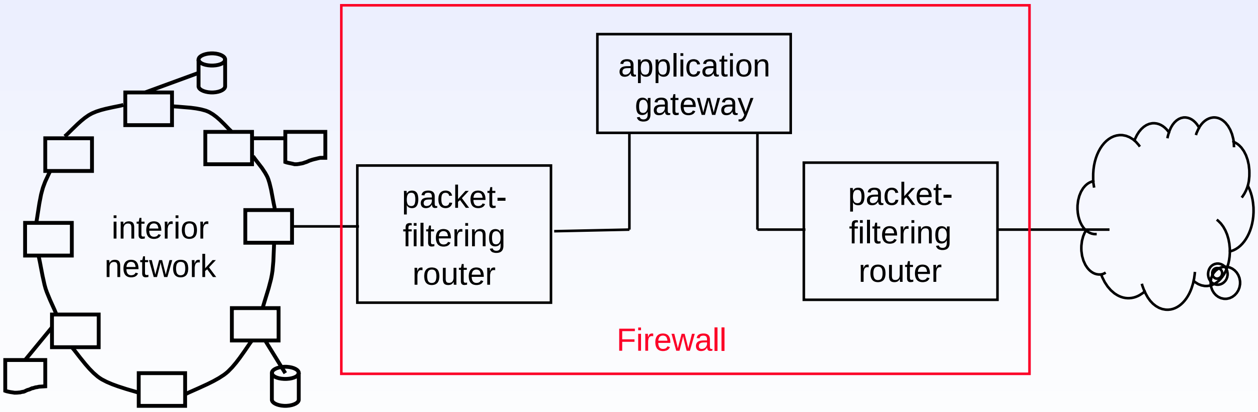
Network layer: internetworking-6

- Tunnelling:
  - source and destination subnets are same type, but intermediate subnets are different
  - use intermediate subnets as big bit-pipe
  - multiprotocol router at source encapsulates end-system packets and then transfers them to the multiprotocol router at destination



Network layer: internetworking-5

- Firewalls:
  – provide a mechanism to control traffic on a protocol and application basis
  – typical firewall contains two routers plus one application gateway
    - one router is on the “inside” LAN
    - other router forms a small outer LAN
    - application gateway routes between inside and outside
  – filtering routers deny/allow traffic based on source & destination addresses, SAPs
  – application gateway filters based on application & packet content



Network layer: examples

- CLNP
  - ISO ConnectionLess Network Protocol
  - ISO 8473-1 (1994)
  - defined in conjunction with OSI reference model, hardly used
- IS-IS & IS-ES
  - ISO 10589 (1992); ISO 9542 (1988)
  - routing protocols for intermediate & end systems
- IP
  - RFC 0791 (1981)
  - an unreliable datagram service
  - addresses are 32-bit integers; for the "Internet", managed globally; private IP networks are common
    - addresses implicitly define network size: the "class a", "class b" etc categories
  - (much) more later

Network layer: examples-2

- X.25
  - standardized by CCITT 1976; X.25 refers to three layers (roughly first three OSI layers)
    - layer 2 is LAP-B (HDLC subset)
    - layer 3 is PLP (packet layer protocol), analogous to OSI network layer
  - connection-oriented services:
    - switched virtual circuit (SVC) – connections created and destroyed as needed
    - permanent virtual circuit (PVC) – connections always exist; like leased line
  - designed for common-carrier (PTT) deployment
    - addresses managed by ITU-T on planetary scope; consist of country, network and address

Network layer: examples-3

- ATM
  - not clear whether the upper layer of ATM is a network layer or not
    - Kurose & Ross seem to think not, since they discuss ATM only in the context of data link
    - Tanenbaum states clearly that ATM **does** represent a network layer, since it switches, routes and provides end-to-end virtual circuits
  - some of the confusion arises from the current practice of running IP over ATM networks
    - if IP is a network layer, and it uses ATM for its services, then ATM must be at a lower level
  - ATM is fundamentally a connection-oriented protocol, providing switched and permanent virtual circuits

The transport layer

- Purpose: provide reliable data transport from source computer to destination computer, independent of network topologies
  - to improve the quality of service provided by the network layer (value-added network)
  - provide a variety of service options to client processes
- If network layer is characterised as modelling a real (physical) (error-prone) network or subnet, the transport layer is an idealised “perfect” network
- The vast majority of transport users want reliable connections, and are not interested in knowing about transient errors
- We will describe general principles here, defer detailed examples to TCP/IP

Transport layer: design issues

- Two services:
  - connection-oriented
  - connectionless
- Same as network layer – what's the difference?
  - network layer is part of the subnet, often run by an external agency (carrier)
  - transport is part of the host
  - if the network layer does not provide a reliable service, the transport layer can
- The transport layer is the interface between the host and the subnet
  - transport layer allows transport layers to be independent of the subnet service variations
  - if networks layers were flawless and provided consistent services, transport layers wouldn't be necessary – maybe!

Transport layer: quality of service

- Transport layer offers "quality of service" (QoS) improvements:
  - generally refers to quality of connections
  - usually given by transport user as parameters to transport service
- Some QoS parameters:
  - connection establishment (CE) delay, CE failure probability
  - throughput, transit delay
  - protection, priority, resilience ("crash" probability)
  - residual error ratio
- QoS parameters are negotiated between transport layer and user
  - desired and minimum levels
  - TL can reject connection if QoS is unrealistic; otherwise contact peer TL to negotiate terms of connection

Transport layer: service primitives

- Services primitives define how to use the services offered by a layer
- Example transport layer services:
  - listen: willingness to connect
  - connect: initiate a connection
  - send: transfer data
  - receive: wait for data
  - disconnect: terminate this side
- TL services usually accessed via standardised operating-system features:
  - programming libraries
    - Unix, Windows socket library for TCP
  - object libraries
  - high-level abstractions, e.g. middleware

Transport layer: protocols

- Transport layer protocols (between peer transport entities) must handle:
  - error detection and management
  - flow control
  - TPDU (Transport PDU) fragmentation, sequencing
- Similar to data link layer, but with differences:
  - DL addressing is easy: the “other end of the wire” or broadcast
  - TL requires complex connection management; in DL, “other end” is always (usually) there
  - subnet (below TL) has storage: datagrams can bounce around for a while; DL frames either arrive or disappear

Transport layer: addressing

- Transport layer users must specify complete addresses of peers:
  - generic: TSAP – transport service access point [also NSAP for NL]
  - how do applications determine a TSAP?
- Several techniques:
  - “well-known” TSAPs
  - process server: accepts connections for many TSAPs and creates peer processes dynamically (Unix–TCP initial connection protocol)
  - name server: service-name-to-TSAP mapping
- May also need TSAP-to-network mapping:
  - hierarchical TSAP address: TSAP is structured so that it contains NSAP and port
  - TSAP flat address space: requires second mapping

Transport layer: connection management

- Transport layer protocols must provide connection creation and destruction (between peer transport entities)
- Consider connection problem:
  - sender S to connect to receiver R
  - S sends a connect request (CR_1)
  - R replies (Rep_1), but message is lost
  - S times-out, resends (CR_2)
  - R thinks it's a new CR, replies (Rep_2)
  - two replies for one pending CR
- But, suppose Rep_1 is just slow, not lost
- Complex problem, cause primarily by ability of subnet to "store"
- Solution: *three-way handshake* (Tanenbaum 6.2.2)

Transport layer: connection management-2

- Releasing (terminating) a connection is also a problem
- Two kinds of release:
  - symmetric: closing from one side does not affect the other direction
  - asymmetric: closing either end terminates both directions immediately
- Asymmetric is abrupt (uncontrolled) and can cause data loss
- Symmetric is controlled, but may need a protocol like:
  - H1: “I’m done. Are you done?”
  - H2: “Yes I’m done. Releasing…”
- Provably impossible to construct such a protocol: the *two-army problem*
- Solution: use a three-way handshake with auxiliary timers (Tanenbaum 6.2.3)

Transport layer: other protocol issues

- Transport layer protocols must also manage:
  - flow control and storage management:
    - e.g. sliding-window
    - can cause significant buffer requirements
  - multiplexing:
    - upward: combining many TL connections into one NL connection
      - optimises uses of subnet (reduce cost)
    - downward: using many NL connections for one TL connection
      - improve effective throughput (assuming different routes)

Session layer

- Allegedly: value-added transport
- Original concepts:
  - dialogue control (synchronization of duplex conversations): whose turn is it to send?
  - checkpoint-restart facilities: e.g. print-spoolers or large-file transfers
    - a file transfer application might structure a transfer as a session that checkpoints from time to time
    - if a transport connection is lost, the session can create another and resume from the last known checkpoint
  - “activity management”: analogous to transaction management (sequence of messages form an atomic activity)
- Not used by TCP/IP reference model
- Possible example: NetBIOS over TCP is a transport-using layer, so it is in the position of a session layer

Presentation layer

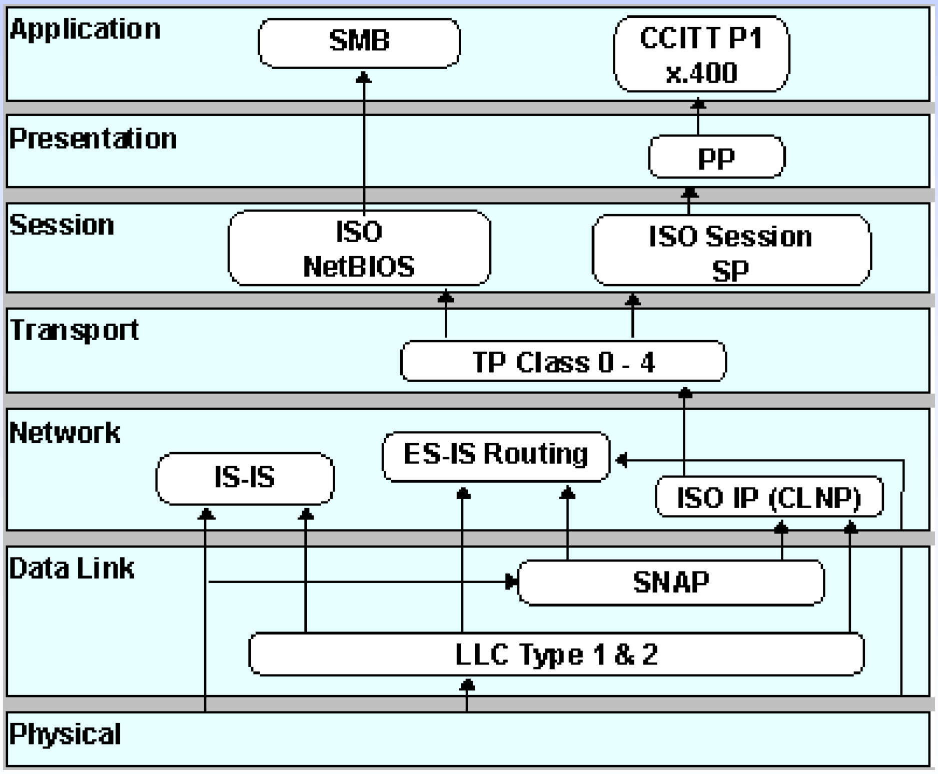
- Allegedly: value-added session
- Practically: non-existent
- Better called the “representation” layer, deals with:
  - character-set translations
  - hardware architecture influences: endian, integer & floating-point, structured storage (e.g. C structs)
  - encryption
  - compression
- Useful notation: ASN.1 – abstract syntax notation
  - used by SNMP (network management)
  - machine-independent way to represent structured data; a *transfer syntax*

Applications layer

- OSI envisaged a huge suite of applications:
  - e-mail: MOTIS, CCITT X.400
  - directory services: X.500
  - FTAM: file transfer, access and management
  - virtual terminal services (similar to telnet)
  - etc
- Didn't happen, except for governments
- De facto standards (mostly TCP/IP, but also some vendor-specific) emerged instead

Summary

- ISO OSI: 7 layers, some good, some not so good
- Originally designed as a complete end-to-end system
  - we would have “OSI cards” instead of Ethernet cards
  - LANs would be OSI-based, telcos would provide OSI WANs



Reference: http://www.protocols.com/pbook/iso.htm