Important terms in Datacom

Network structures

Components in Network

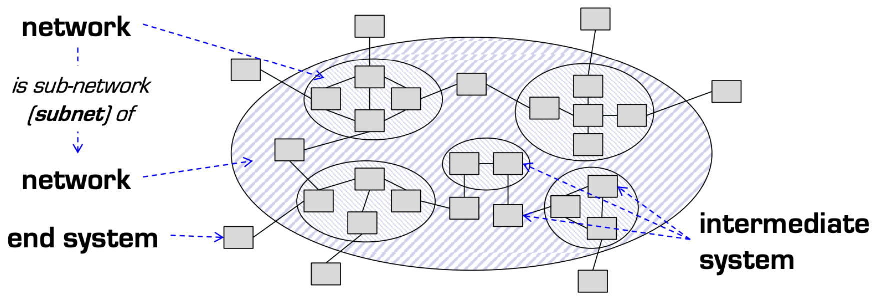


Figure 1: Illustartion of components in Network

End System (ES):

In the end/edge of a network.

- Phone
- Car
- PC

Intermediate System (IS):

Is inside the network, often a guide for the ES.

- Router
- Switch
- Proxy server

Structures in Network

Point-to-point channel

When a system is trying to reach out to a spesific system.

- Star topology (most common)
- Tree topology
- Ring topology
- Full mesh (super computer)
- Fat tree (super computer)
- Torus (super computer)
- Hypercube (super computer)

Broadcast channel

When a system is trying to reach out to many as possible.

Networks tasks

- Knowing how to find the reverse way
- Knowing which process to contact on that ES
- Knowing which ES to contact
- Knowing the way from a IS to another
- Coding the data in a comprehensible manner
- Maintain privacy
- Maintain security
- Avoid delays
- Support hight traffic
- Dealing with network problems

Structuring the task

Layered approach

- Arrange task in layers
- Clear interface
- Clear assignment of responsibilities
- Not perfectly suited for all jobs
- Similar problems are solved several times

Component approach

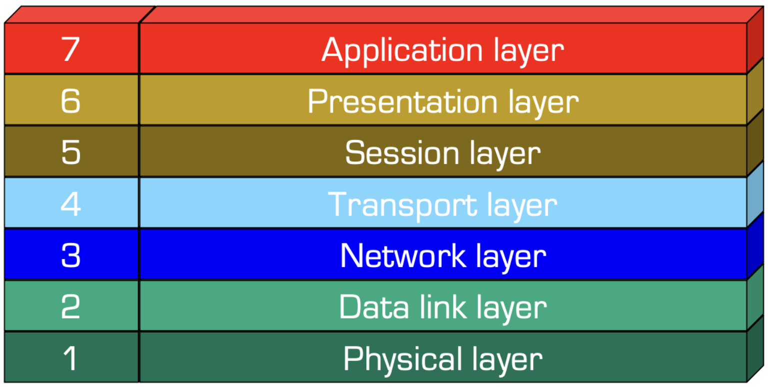
- Interacting components
- Possible to avoid duplicated functions
- Possible to choose perfect network behaviour for every application
- Must negotiate choice of every piece with all nodes
- Toolbox must be complete on all nodes
- Needs flexible interfaces

Recursive approach

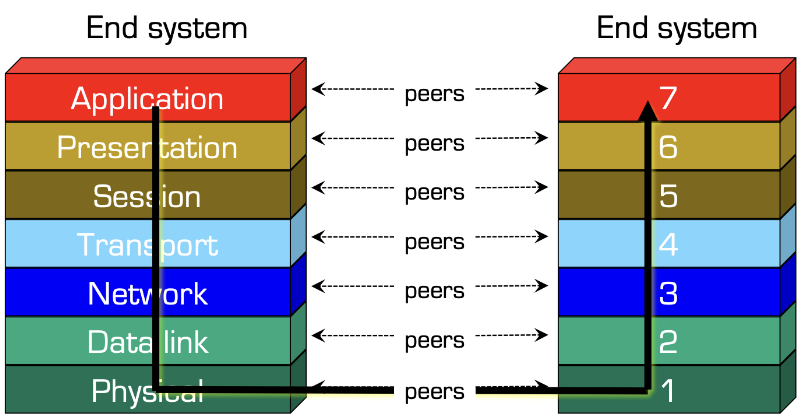
- Handle challenges locally
- Reuse the concept of interprocess communication on all levels
- Concepts are repeated at every level
- All challenges can be solved as local as possible
- More negotiations and setup than layered
- Unclear how to best share resources

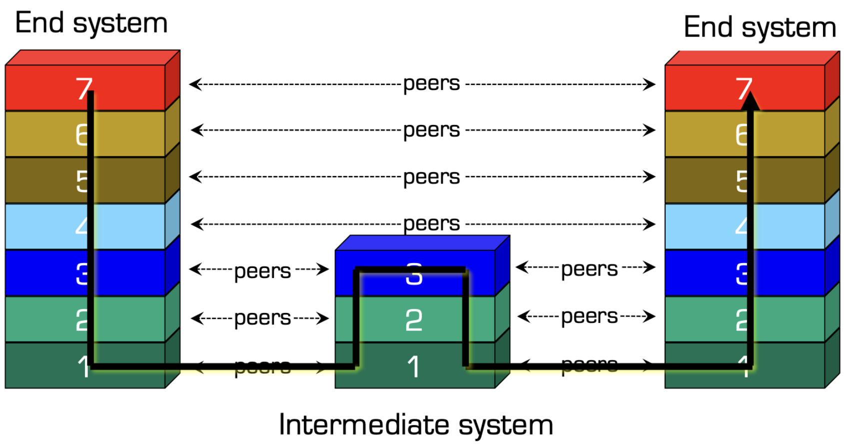
Layering model

ISO OSI (Open Systems Interconnection) Reference Model

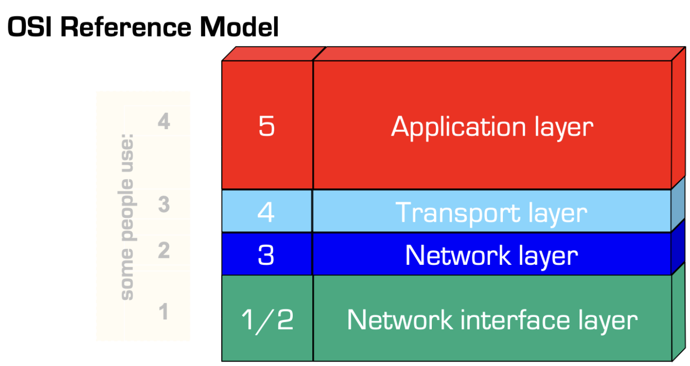


Architecture

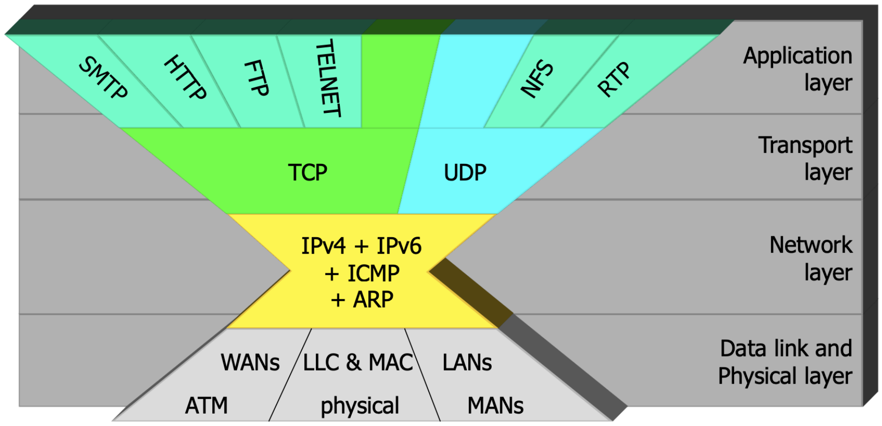




TCP/IP Reference Model Internet Architecture



Internet Protocol Stack



Naming in datacom

Protocol: Communication between same layer

A protocol defines the format, the order of messages. A protocol often exchange between two or more communication entetites.

Network byte order

Big Endian vs. Little Endian

Representation of numbers

Example: The decimal 36
In binary: $1 \times 32 + 0 \times 16 + 0 \times 8 + 1 \times 4 + 0 \times 2 + 0 \times 1 = 100100$
In hexa: 0x24

```
                       00100100
⇔                    0010 : 0100
⇔ 0*8 + 0*4 + 1*2 + 0*1 : 0*8 + 1*4 + 0*2 + 0*1
⇔                       2 : 4
⇔                        0x24
```

Big Endian

The most significant byte reads first. Most common way to read.

Little Endian

The least significant byte read first. Easy to transform

Addressing (MAC addressing)

MAC address = Task is to identify different systems that uses same local network

Point-to-point channels

- MAC addresses is not required in this network structure

Broadcast channels

- MAC address is important in a true broadcast channel
- MAC adress has only local meaning, therefor will nodes in the “other side” of IS not know them

Layer 3 Addressing Resolution

Direct Mapping

- The 32 bits IP-address would fit into the 48 bit destination MAC address
- But you have to change the destiantion for every IS because MAC addresses is unique

Mapping table

- Each node has a table for transfer from IP address to MAC adress for their neigbours in layer 2 (data link layer)
- Forced to update tabels

Address Resolution Protocol (ARP)

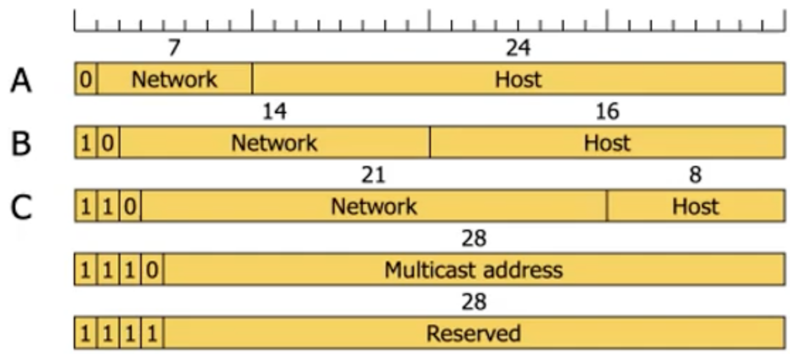
- Used to resolve IP addresses to MAC addresses
- If the local machine thats sending a packet has IP address in their cache, the packets sends. If not, a request of who hsa the IP address on broadcast address. We get the receivers MAC address and with ARP, the IP adress mapps to MAC address.

Reverse Address Resolution Control

- Skrive

Addressing (IP-address)

Internet Addresses and Internet Subnetworks



Address Class	RANGE	Default Subnet Mask
A	1.0.0.0 to 126.255.255.255	255.0.0.0
B	128.0.0.0 to 191.255.255.255	255.255.0.0
C	192.0.0.0 to 223.255.255.255	255.255.255.0
D	224.0.0.0 to 239.255.255.255	Reserved for Multicasting
E	240.0.0.0 to 254.255.255.255	Experimental

CIDR: Classless InterDomain Routing

Finds the right route by choosing:

- The router with longest mask (highest number last)
- And have identicall bits "before" netmask in IP-address

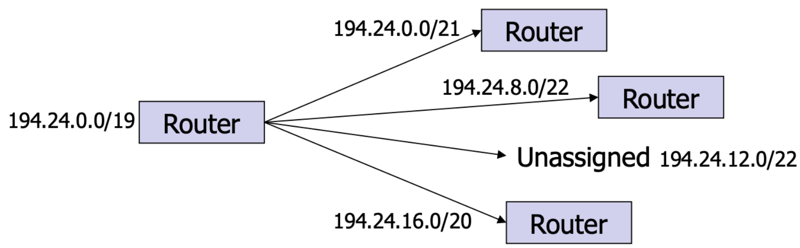


Figure 2: Example

Layer 3 addressing (IPv6 addresses)

IPv6

Problems with IPv4

- Too few addresses
- Bad support for QoS (quality on service)
- Bad support for mobility
- No IP-addresses for individuals

Pros with IPv6

- Support billions of end system, because of longer addresses
- Reduce routing table
- Simplify protocol porcessing (header)
- To increase security
- IPv6 is larger, but more simple

Layer 4 Address Resolution

Transport Layer Functions

1. Addressing
2. End to end communication
3. Transparent data transfer between processes
4. Quality of service options (QoS)

- Error recovery
- Reliability
- Flow control
- Congestion control

Transport Layer Addressing

1. Port
2. IP address

UDP

Connectionless

- No error recovery
- No reliability
- No flow controll
- No congestion controll

TCP

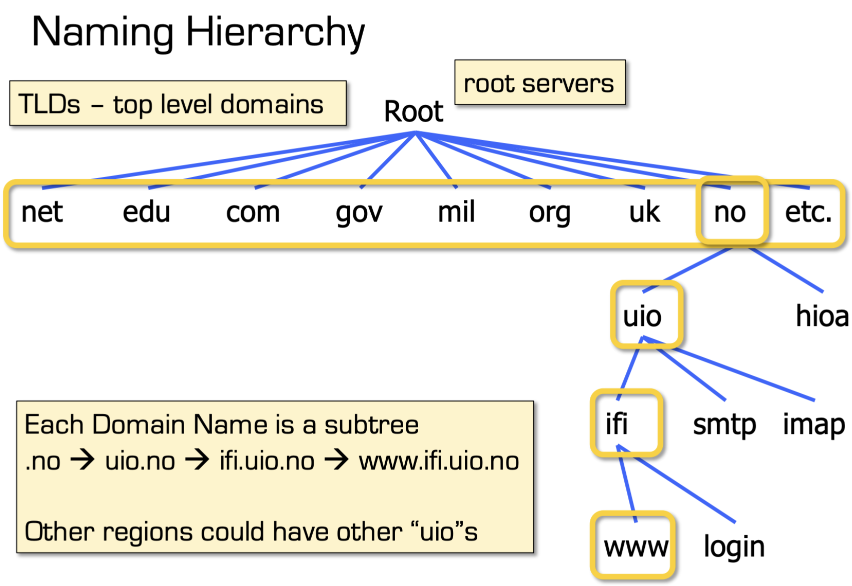
Connection-oriented

- Error recovery
- Reliability
- Flow controll
- Congestion controll

Layer 5 Address Resolution

Domain Name System (DNS)

Like a phonebook for the internet! When you want to visit a website, you type in a web address (like www.google.com) which is easy for people to remember. However, computers on the internet locate websites using IP addresses (a series of numbers like 192.0.2.1).



Recursive DNS Query

Classical approach

- Must keep state for every request in a server until answered
- Allows every node along the path to cache results
- Concentrates the data flow at the central servers
- Keeps a lot of state on central servers

Iterated DNS Query

Newer approach

- Redirect request
- Keep state only at local server (or some servers) until answered
- Allows few nodes to cache results
- Halves number of requests at central servers
- Avoids state on central servers entirely

End-to-end delivery on layer 3

Network Layer

Primary task from a layer model perspective

- Connectionless or connection-oriented service

- Uniform addressing
- Internetworking: provide transitions between networks
- Routing
- Congestion control
- Quality of Service (QoS)

Inside a Network Layer

Routing and Switching: Terminology

Routing

When an IS reads a destination address from an arriving packet, computes which of its direct neighbors is best suited for reaching that destination, and sends the packet to the neighbor.

Switching

When an IS reads an identifier from an arriving packet, looks it up in a pre-filled mapping table that translates the identifier to a direct neighbor, and sends the packet to the neighbor.

Circuit Switching

Connection exists physically for the duration of the conversation

- Switching centers
- Connections between switching centers
- Estabilishing a connection takes time
- Once a connection is established it cannot be blocked anymore
- Connection establishment can take a long time

Virtual Circuit Switching

Setup path from source to destination for entire duration of call

- Using state information in nodes but no physical connection
- Connection setup: defines data path

Message Switching

- All data to be sent are treated as a "message"
- High memory requirements at the node (switching centers)
-

Packet Switching

- Packets of limited size

- Dynamic route search (no connect phase)
- No dedicated path from source to destination
- Possibly only reservation of average bandwidth (static reservation)
- Possibility of congestion
- High utilization of resources
- No connect phase
- No allocation of bandwidth

Flow Control on Transport Layer

Flow controlls task: Make sure the sender is not sender more data than the receiever can receive.
Problem: Sende sends data faster then the receiver can receive.
Without flow control: Data get lost.
With flow control: Sender can adapt to receivers abilities by feedback.

Stop & Wait

A flow control mechanism where you
A) Fix the loss
This means that you have a timer from the sender when the sender sends data. The timer will stop when they receive an ACK.
B) Distinguish packet loss and ACK loss
We can both loose a packet and an ACK, but to get control over which packet or ACK we lost, we use SEQNO which is singel bit with either 0 or 1 value.

Sliding Window Consept

Stored packets at the sender

- the sender must store all unacknowledged packets and be able to retransmit
- maximum number defined by sender's window size (here 3)
- the packets not yet acknowledged by the receiver

Stored packets at the receiver

- not necessary to store any packets
- not useful to store more than one receiver window size

ACK sent by receiver only if the packet

- has been identified as being correct
- can be transmitted correctly to the application

Two windows (send + recv) which is the buffer. The sender sends more packets at the same time, and the receiver receive the packets and sends ACK back to the sender.

The sender must save the packets until thet have received an ACK, incase if lost and needed to send a new ACK. The receiver doesnt need to save the packets, only needs a ref to what the last packet that got ACK, fits right in the queue.

Lower Bounce and Upper Bounce

If LB and UB is the same, where not waiting on something. The window size is sat by the difference between LB and UB.

Senders side

- UB goes up with how many packets that has been sent and LB goues up with ack sendt back
- You dont need to wait to get an ACK from everyone before you send next packet

Receivers side

- LB starts at 0 and UB starts at 3
- LB goes up when a packet is received and UB goes up when sending an ACK
- If LB = UB, buffer is full

Sliding Window: Go-back-N

No buffer allocation.

- All sendt packets has their own timer, the receiver doesnt need to tell if a packet has been thrown away, because the sender understands that packet that has been sent also has been thrown when the timer goes out. Pros: The receiver doesnt need to handle the order of the sent packets.

Sliding Window: Selective Repeat

Static buffer allocation

- The sender sends 3 packets, but one packet dissapear. Receive receives packet 0 and packet 1, sends an ACK 0, but keeps ACK 2 instead of throwing it away.
- Accumulative ack
- Pros: Less traffic

Credit Mechanism

Dynamic buffer allocation

- The receiver send a “credit” to the sender, the sender knows hoe many packets to send. When the receiver sends 4 credits to the sender, the sender knows it can send 4 packets. The receiver makes the buffer ready to receive packets.

Congestion Control on Transport Layer

Congestion can be described as trafic in the network. Too much trafic can congestion controll fix.

We have **Presitent Congestion** where the router stays congested for a long time. And **Transient Congestion** where the router is temporarily overload. And the congestion happens in a certian period. This type of congestion happens because of waves of trafic.

TCP Congestion Control

The original TCP didnt have a congestion control. The TCP we use today:

- TCP New Reno
- TCP Prauge
- TCP BBR
- TCP Cubic
- TCP PPR

Congestion Window (cwnd)

Most data kan be in the network at the same time from a connection

Maximum segment size (MSS)

Most bytes in a TCP entity can be sent together (max 60 bytes)

Threshhold (ssthresh)

A level, point, or value above which something is true or will take place and below which it is not or will not

TCP Tahoe

Threshold sets to half of the congestion window if we loose a packet. Congestion windows sets back to 1.

TCP Reno

Threshold sets to half of the congestion window if we loose a packet. Congestion windows goes back to a new threshold.

Picture

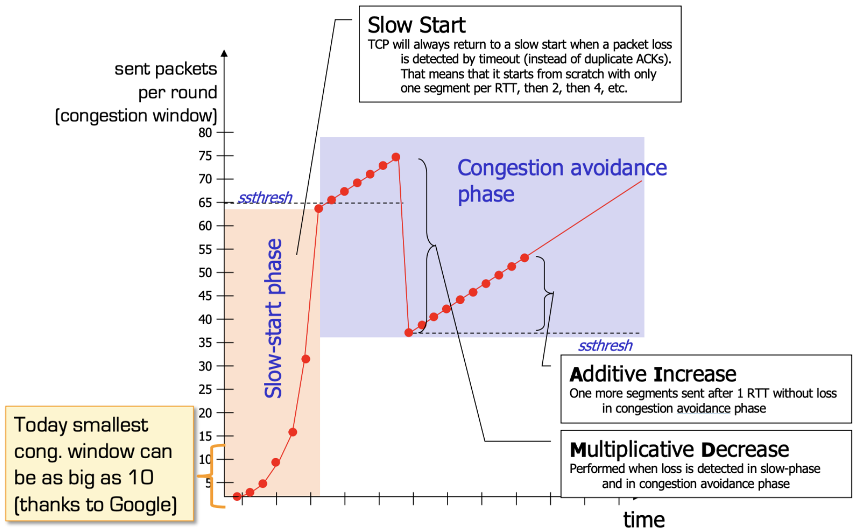


Figure 3: Illustration of how congestion controll works

Routing on Network Layer

The networks layers task: provide service to the transport layer.

Routing: Fondation

- Task
- Routing Algorithm
- Route determination

Good Properties for Routing Algorithms

- Correctness
- Simplicity
- Robustness
- Stability
- Fairness
- Optimaly

Non Adaptive Algorithms

- Static routing (the routing is planned)

- Spanning tree and flow based routing (with knowledge of the topology)
- Flooding (without knowledge of the topology)

Adaptive Algorithms

- Decisions is based on current network state
- Further sub-classification into: (Centralized algorithms, Isolated algorithms, Distributed algorithms)

Conflicts

Fairness and optimaly

Sink tree

Dijkstra

Link State Routing (LSR)

- IS-IS (Intermediate System-Intermediate System)
- OSPF (Open Shortest Path First)

Basic principle

- Every IS measures the distance between their direct neighbour
- Distributes the information
- Calculate the ideal route

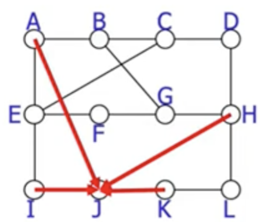
Procedure

1. Determine the address of adjacent IS
2. Measure the "distance" (delay, ...) to neighbouring IS
3. Organize the local link state information in a packet
4. Distribute the information to all IS
5. Calculate the route based on the information of all IS

Distance Vector Routing

Every IS maintains a table (vector) stating.

- Best known as distance to destinations
- ISes updates tables by exchanging routing information with their neigbours



	A	I	H	K		line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	-
K	24	22	22	0	6	K
L	29	33	9	9	15	K

delay	JA	JI	JH	JK
	8	10	12	6

Figure 4: Distance Vector Routing