Imperial College London

Computer Networks and Distributed Systems

Part 4 – Network Layer

Course 527 – Spring Term 2015-2016

Emil C Lupu and Daniele Sgandurra
e.c.lupu@imperial.ac.uk, d.sgandurra@imperial.ac.uk

Part 4 – Contents

Interconnecting networks (Layers 1 to 3)
- Repeaters, bridges, routers

Network Layer
- **Routing**
  - Static, distance vector, link state
- **Internet Protocol** (IP)
  - Datagrams (packets)
  - IP addressing
  - Fragmentation
  - Other protocols (ARP, ICMP)

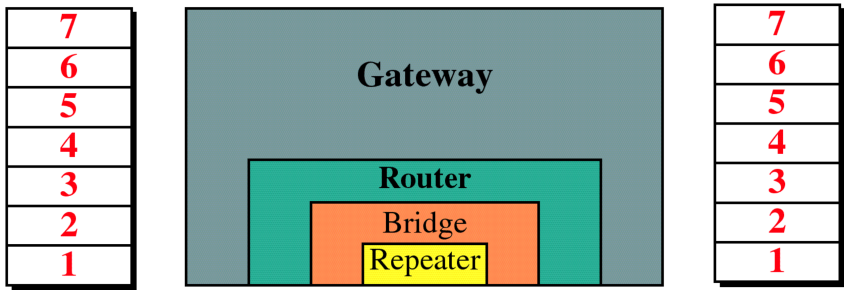
Inter-Networks

Inter-networks formed from smaller networks
- Extending physical limits of networks
- Separating traffic (to spread load or administration)

Different devices interconnect with different low-level protocols
- Cooperation at higher layers to provide uniform service

Connecting Devices and OSI Model

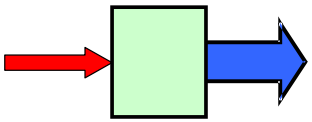
Repeater/Hub
Bridge/Switch



Repeater

Amplifies electrical signal

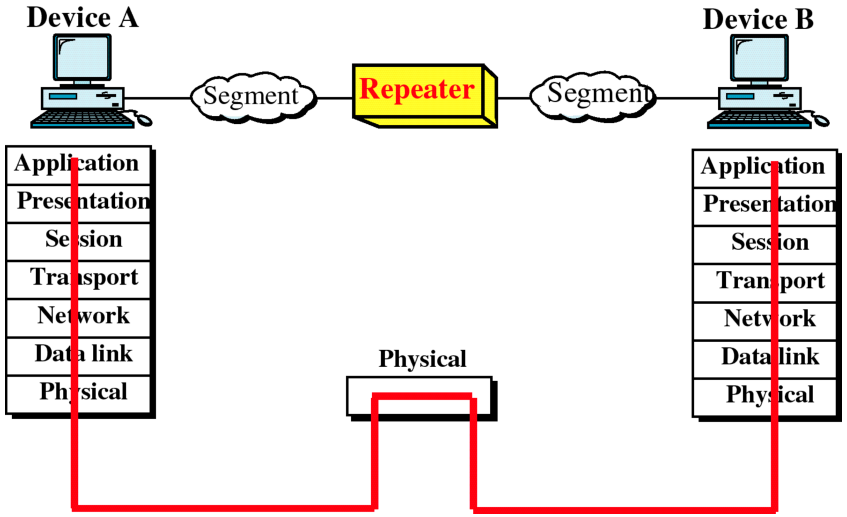
- Makes two wires appear as one
- Improves signal propagation distance



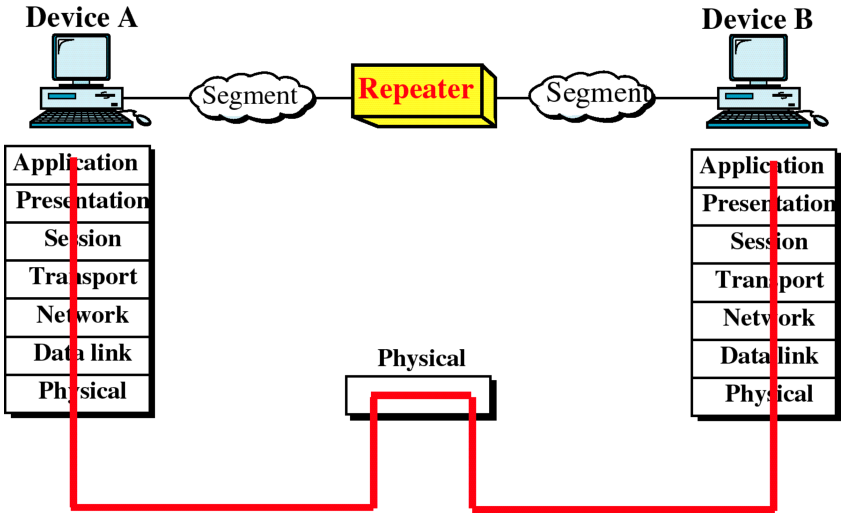
Operates at physical layer

- Transparent to higher layers
- No checking/generating of checksums
- CSMA/CD must cope with longer propagation delays
  - Ethernet (10Mb/s): up to 4 repeaters with 2.5km max length

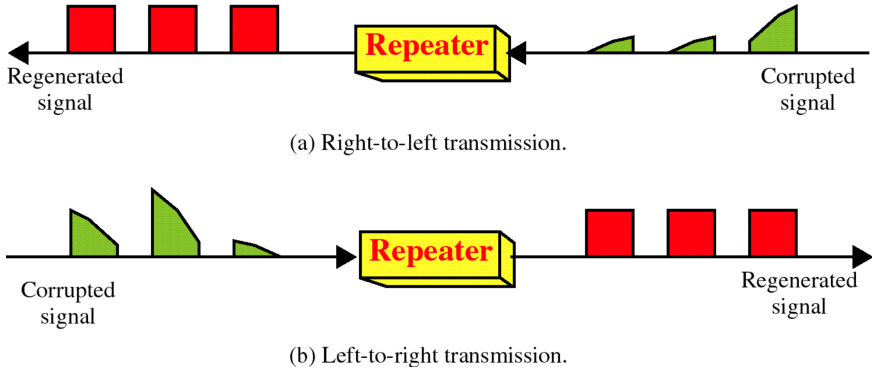
Repeater/Hub and OSI Model



Repeater/Hub and OSI Model



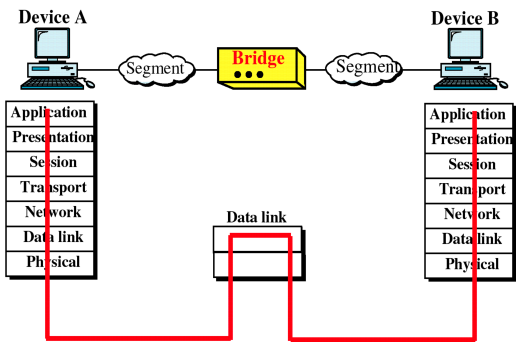
Function of a Repeater



Bridge/Switch

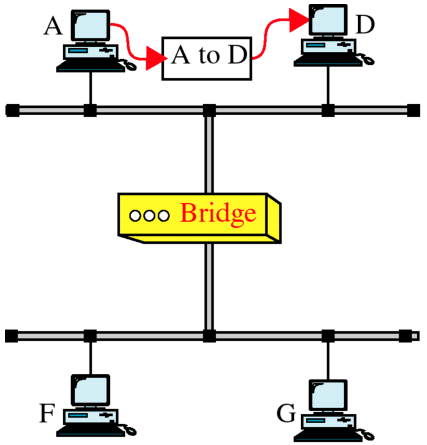
Interconnecting LANs with traffic isolation

Conditional forwarding
- Only forward frames destined for other LAN

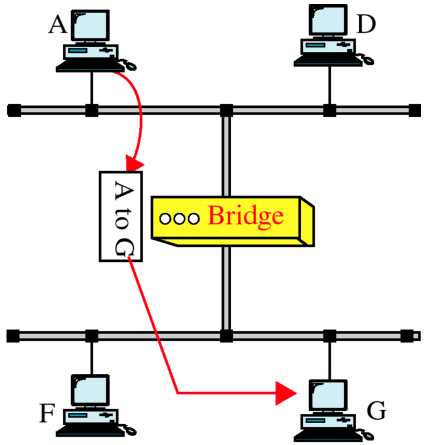


Operates at data link layer
- Reduces load on sub-network
- Store & forward results in higher delay
- Network layers must be same (but not processed)
- Physical layers may be different

Function of a Bridge/Switch



a. A packet from A to D

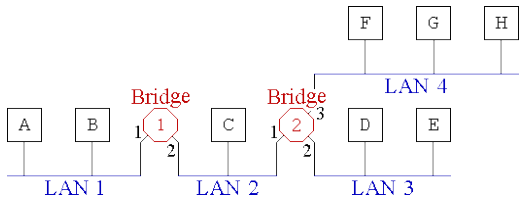


b. A packet from A to G

Transparent (Spanning Tree) Bridge

Bridge records source addrs & links in table

- If destination addr on same link as source ➔ do not forward
- If link for destination addr known ➔ only forward on that link
- Otherwise use flooding
  - Send on all non-source links



Host	A	B	C	D	E	F	G	H									
Link	1	1	2	2	2	2	2										

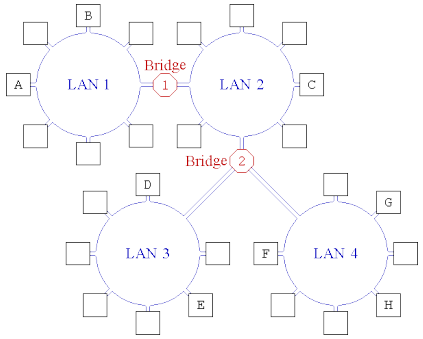
Backwards learning
- Over time all hosts should send frames
- Creates complete host/link tables

Loops in topology
- Make determining location of source impossible
- Causes frames to proliferate

Network layer protocol often handles loops
- Packets may have limited lifetime

Build spanning tree (loop-free subset)

Source Routing Bridge



Bridge issues **discovery frame**
- Copied down every link, recording path list

Destination chooses route based on discovery frames
- Or, sends discovery frames back to sender for decision

Routing path carried in data frames
- Connection-oriented

Keeps bridges simple but end hosts complex
- Hosts must discover routes and putting routes in frames

Route exploration can wipe out benefits
- Bad for networks with high degree of connectivity
- Must cache routes or be very inefficient

Have to rerun discovery if bridge/route fails

Token Ring networks use this

Comparison: Types of Bridges

Transparent	Source Routing
Connectionless	Connection-oriented
– Low overhead to send one frame	– Overhead of discovery on first frame
– Failures handled by bridge	– Failures handled by hosts
Transparent at hosts	Not transparent at hosts
– Backwards learning location of hosts	– Discovery frames locate host
Sub-optimal routing	Optimal routing
Complexity in bridge	Complexity in hosts

Combination: Mixed Media Bridge

Interconnect different networks
- e.g. Ethernet and Token Ring
- Can be source-routing/transparent on different sides
- Holds routing tables which differentiate network type

Handles different **maximum frame lengths** (segmentation/fragmentation)
- 1518 Bytes on Ethernet
- 4KB on 4Mb/s token ring
- 17.6KB on 16Mb/s token ring

Fast Bridges/Packet Switches

Packet switching for LANs
- Since 1984 (DEC were first)
- Switches with one port per LAN
- Reduction of collision domains

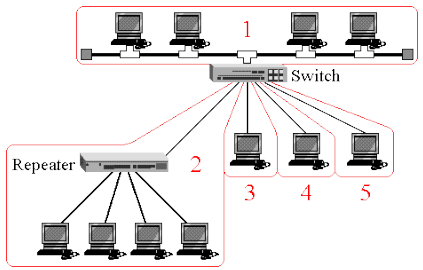
Modern switch technology has improved
- Multi-port bridges forward frames between all ports at wire speed
- Don't use store-and-forward but rather cut-through
  - Forward as soon as dest header field received

Switches are not Hubs

Switched LANs
- Looks bit like shared-bandwidth LAN with hub
- One network cable per computer

But different in terms of:
- frame propagation
- congestion
- cost

Separating Collision Domains



Shared medium requires CSMA/CD to arbitrate
- Contention can be problem on busy network
- Hosts in separate collision domains not competing for media

Switches form ends of **collision domains**
- Reduces to collision domains of 2 (switch + host)

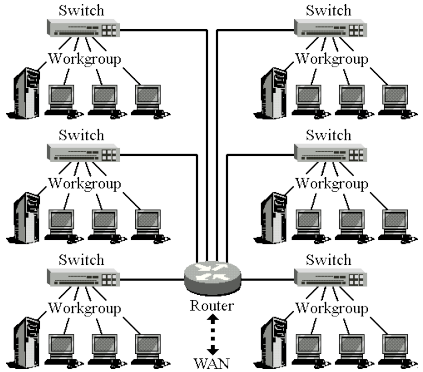
Separating Data Rates

Devices on shared LANs operate at same data rate

Distinct LANs may operate at different rates
- Ports of switch can operate at different rates
- Connection between two hosts limited by slowest hop

Common configuration
- Switch interconnects run much faster than most hosts
- Some hosts have high performance links

Segmentation with Switches



Switches can segment traditional networks

Collapsed backbone

- Backbone in switch rather than shared wire

Hub, Switches vs Routers

Network Switch

- Lives at Datalink layer
  - Knows about MAC addresses and frame formats
- Interconnects network segments

Hub

- Lives at Datalink layer
  - Knows about MAC addresses and frames
- Passively interconnects ports ➔ acts as single network segment

Router

- Lives at Network Layer
  - Knows about IP addresses and IP packets
- Interconnects separate networks
- Carries out more intelligent routing decisions

Network Hubs and Switches

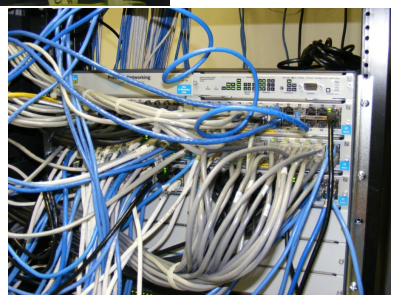
Network Hub

Not on sale anymore?

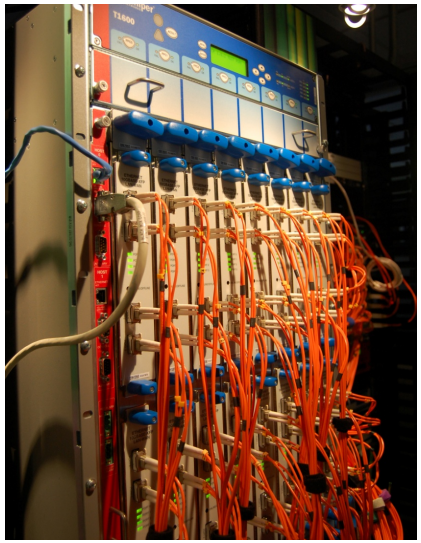
Network Switch

Ethernet Gigabit Switch

- 48 ports
- Cost: ~ £1500



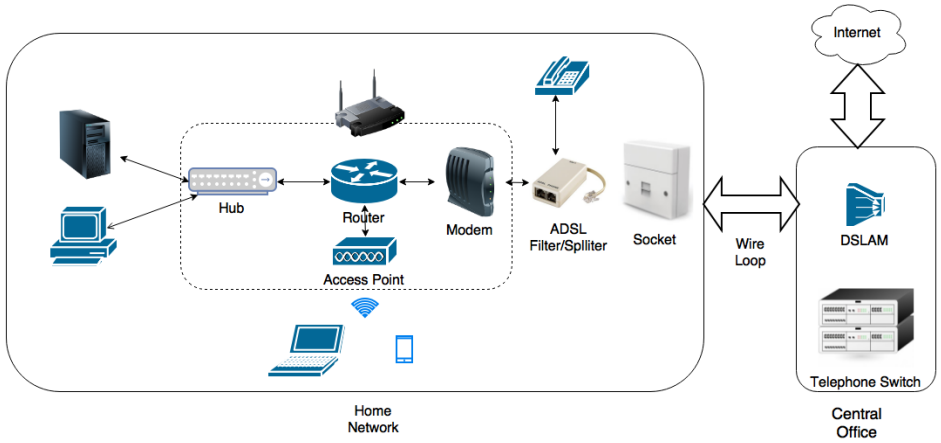
Network Router



Juniper T1600 Core **Router**

- Routes 2 billion packets per second
- 1.6 Tbps capacity
  - 160 ports with 10 Gbps
- Cost: ~ $300,000
- Possible to interconnect up to 16 of them ➔ 30 billion packets per second

Home Network



Routing

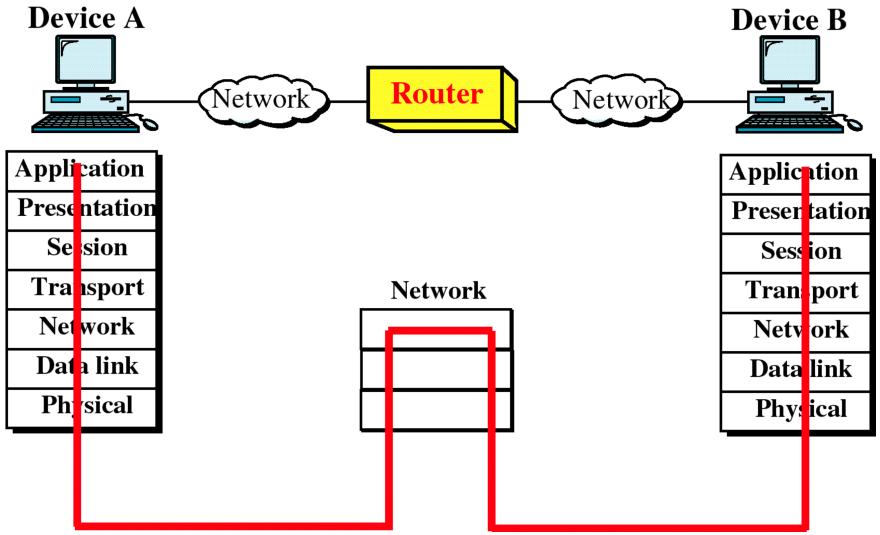
Problem: No single network can serve all users

- Network too long, too much traffic, too complex for lower layers, can't maintain complete network plan
- Think Internet scale!

Solution:

- LANs (subnets) interconnected using **routers**
- **Routing** refers to selecting path from source to destination across multiple subnets
- Network layer must cope with differing underlying LANs

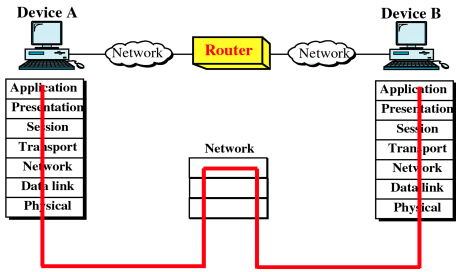
Router and the OSI Model



Router/Gateway

Router

- Determines next hop for packet, depending on dest addr
- Lookup in routing table



Operates at network layer

- Router forwards packets based on dest networks
  - Unlike bridges, which use hosts
- Verifies/modifies packets
  - Updates fields affected by routing
  - Checks/recalculates checksum

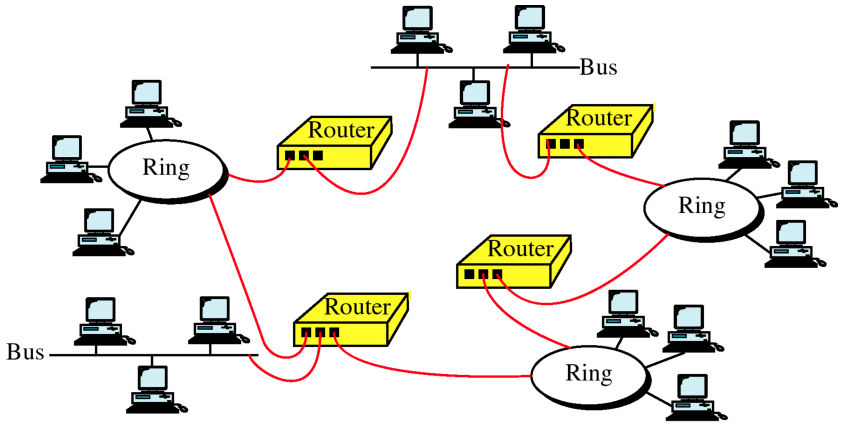
Typically used for connecting sites

- Overcome physical and administrative boundaries
- Greater management and traffic isolation

Not transparent to end nodes

- Frames addressed to router's data link address
- Host needs to know whether/which router to send to

Routers and an Internet



Routing: Objectives

Correctness: Find a route (if it exists)

Efficiency: Routes should provide good performance

- Routes should use minimal resources

Robustness: Return route even when links/nodes fail

Fairness: Hosts should have equal access to network

- Respect priority markings for Quality of Service (QoS)

Adaptability: Routes should reflect network conditions

- But no overreacting to problems

Simplicity: Cheap, predictable and verifiable

Routing: Metrics

Efficiency: Find routes with good properties in terms of

- available bandwidth
- delay
  - Link latencies
  - Hop count
- price
- priority for traffic types

Routing: Properties

No centralised control
- No knowledge of topology or underlying protocols

Interconnection on global (Internet) scale
- May use intermediate networks to get to destination
- Hide underlying interconnection of networks from users
- Networks may be not completely inter-connected

Routing Strategies

Static (non-adaptive) routing
- Compute routes once and load into router
- Worked for early ARPANET

Dynamic (adaptive) routing
- Change routes to reflect changes in topology/load (as seen through congestion)
- Usually used in packet-switched networks
- **Distance Vector Routing** and **Link State Routing**

Non-Adaptive Routing: Static Routes

Routing using fixed directory
- Full address maps to route to host
- Default link for unknown hosts

All packets for host pair always take same route

Often used with list of known hosts/links
- May be set up by pathfinder algorithm (similar to source-routing bridge)

Static routing tables for workstations use this
- Most traffic sent to default gateway/router

Adaptive Routing: Flooding + Random

Flooding
- Send packet to all neighbours except source
  - Unless packet seen before (remove loops)
- Shortest path and fast discovery
- Good for pathfinders and essential/low latency data
- But inefficient and leads to high load on network

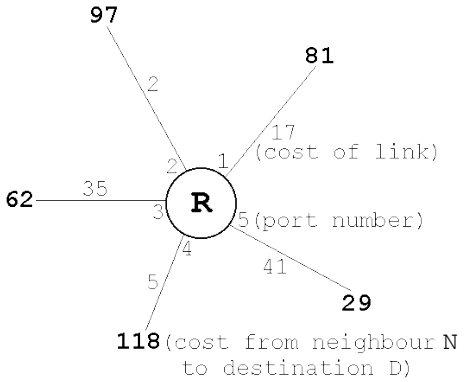
Random
- Forward packet to random link
- Highly robust but slow convergence and inefficient

Adaptive Routing: Distance Vector

Used in ARPANET and Internet until 1979
- By Bellman-Ford, Ford-Fulkerson
- Implemented as Routing Information Protocol (**RIP**)

Router maintains table (vector) of distances
- Usually delay/queue length to each neighbour
- Periodically exchanges this information with neighbours
- Re-computes distance and updates its tables

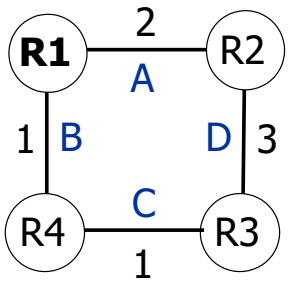
Example: Distance Vector Routing



cost (R → D) =
cost (R → N) + cost (N → D)

Port 1 → 17 + 81 = 98
Port 2 → 2 + 97 = 99
Port 3 → 35 + 62 = 97
Port 4 → 5 + 118 = 123
Port 5 → 41 + 29 = 70

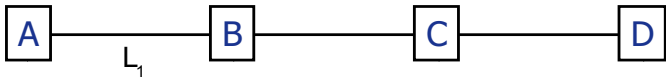
Best choice here is port 5, with distance vector of 70



Step 3

	To R1	R2	R3	R4
Vector R2	2	0		4
Vector R4	1	3		0
Routing Table R1	0 -	2 A	∞ -	1 B

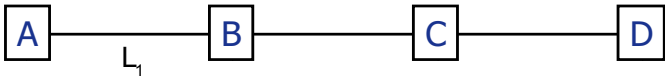
Tutorial Question: Distance Vector Routing



Link 1 is down

	A	B	C	D
A	0	∞	∞	∞
B	∞	0	1	2
C	∞	1	0	1
D	∞	2	1	0

Tutorial Question: Distance Vector Routing



Link 1 comes up

Time	B	C	D
0	∞	∞	∞
T	1	∞	∞
2T	1	2	∞
3T	1	2	3

Link 1 goes down

Time	B	C	D
0	1	2	3
T	3	2	3
2T	3	4	3
3T	5	4	5
4T	5	6	5

Good news travels fast. Bad news travels slowly

- "Counting to infinity" problem. Because e.g., B has no means of knowing it is on the path that C advertises.

Distance Vector Problems

Poor efficiency

- Slow to converge after changes
- Distance vectors increase linearly with network size
  - May not fit inside packet

Route finding suboptimal

- Only considers delay not bandwidth of links
- Prone to oscillations in cost
  - Routing tables do not include paths

Adaptive Routing: Link State Routing

Properties

- Faster convergence and more reliable
- Less bandwidth intensive than DVR
- But more complex and memory/CPU intensive

Each router maintains (partial) map of network

- Consists of more than just neighbours
- May include bandwidth and other metrics

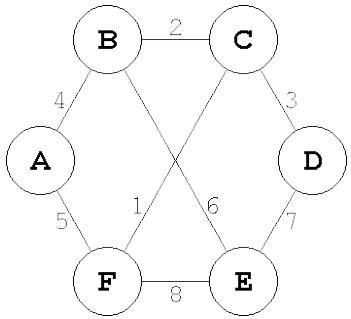
Each router does the following:

1. Discover identities of all neighbours
2. Measure delay (or cost) to neighbours (ECHO packet)
3. Construct and send Link State packet to all routers
4. Compute shortest path to every other router
   - Use Dijkstra's algorithm

When link state changes

- Notification packet flooded throughout network
- All routers re-compute routes

Link State Packets



A	
Seq No	
TTL	
B	4
F	5

B	
Seq No	
TTL	
A	4
C	2
E	6

C	
Seq No	
TTL	
B	2
D	3
F	1

D	
Seq No	
TTL	
C	3
E	7

E	
Seq No	
TTL	
B	6
D	7
F	8

F	
Seq No	
TTL	
A	5
C	1
E	8

Link state packet

- ID of source, sequence number (to handle order & loss)
- Time-to-live (decremented each second until discarded)
- List of neighbours with costs

Link State Distribution

Based on flooding algorithm

- Don't send on incoming link

SeqNo to ensure only newer state packets forwarded

- Drop old & duplicate packets
- Some delay in forwarding to wait for newer packets

Different routers have different views of topology

- Inconsistencies, loops, unreachable nodes

Hierarchical Routing

Complete Internet map in every router infeasible

Instead exploit hierarchy and use regions

- Router knows local topology in detail
- Router knows route to other regions
  - But not their internal arrangements

Regions may map to:

- Geographical area (e.g. London academic network routes between universities)
- Organisation's network (e.g. Imperial has routers in core network, routing between departments and to external links)

Internet Routing

Autonomous systems (AS) are regions on the Internet

Within ASs: Open Shortest Path First (**OSPF**)

- Variant of Link State Routing
- Supports load balancing over multiple lines
- Routing includes type of service (but not used)

Between ASs: Border Gateway Protocol (**BGP**)

- Variant of Distance Vector Protocol
- Records exact path used
- Supports custom routing policies

Summary: Network Interconnection

Repeater (Physical layer)
- Extends range of signals

Bridge (Data link layer)
- Segments collision domains, transparent or source routing

Switch (Data link layer)
- Separates networks, wire speed bridge with multiple ports

Router (Network layer)
- Interconnects LANs, LAN not host addressing, visible to end nodes, needs routing protocol

Internet Protocol (IP)

Basic protocol for the Internet
- Defined in RFC 791

Datagram oriented
- Treats packets independently
- Packets contain complete addressing information
- Unreliable delivery (no notification)
- Variable sized data payload
- No checksum on data payload, just on header

IP Services

Addressing

Packet timeouts
- Avoid congestion and routing problems

Fragmentation
- May split packets if underlying network requires it

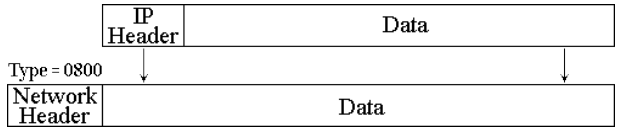
Type of Service through priorities
- Requires routers on path to read and treat differently

Other options
- Source routing requirements, route recording, security labels

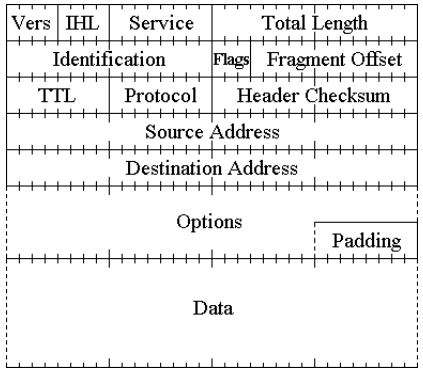
IP Datagrams

IP datagrams are "virtual" or "universal" packets
- IP dest addr is always final destination address
- Physical dest addr in frame is changed at each hop

Along the path each router:
- Removes packet from LAN frame
- Determines next router/local link
- Re-encapsulates in appropriate LAN frame for next hop



IP Datagram Format



Version: IP version (usually 4)

Internet Header Length
- In 4 byte multiples (5<=IHL<=15)
- Options increase this
- Gives data offset

Type of Service
- Trade-off between delay, reliability and throughput

Total Length
- max 64KB with IPv4

Time to Live (TTL): Handles routing loops
- Decremented each routing hop
- Datagram dropped when = 0

Protocol
- 0 = reserved, 1 = ICMP, 6 = TCP, 17 = UDP
- Similar to Ethernet protocol type field

Header checksum: 1's complement sum of header, not data
- Sum of header and checksum should = 0

Source and destination addresses

Options
- Security, loose/strict source routing, record route, stream ID, timestamp, …
- Padded to multiples of 32bits

IP Addressing

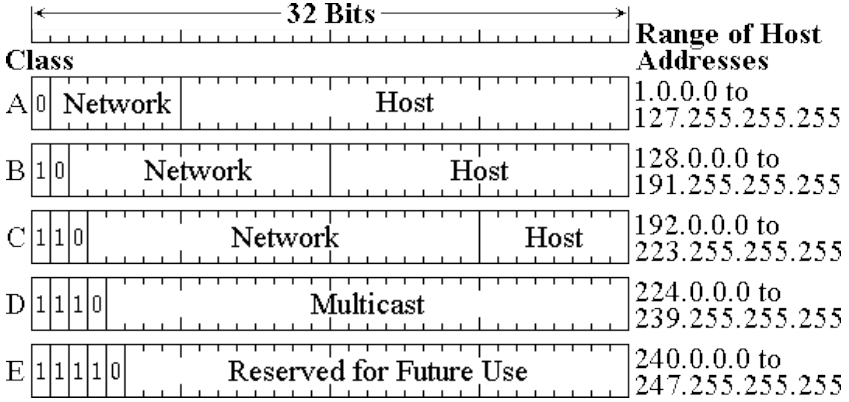
Ethernet addr 48 bits, written as hex pairs

IP addr 32 bits, written as dotted decimal
- e.g. `146.169.7.41`
- No direct mapping of IP addrs to Ethernet addrs

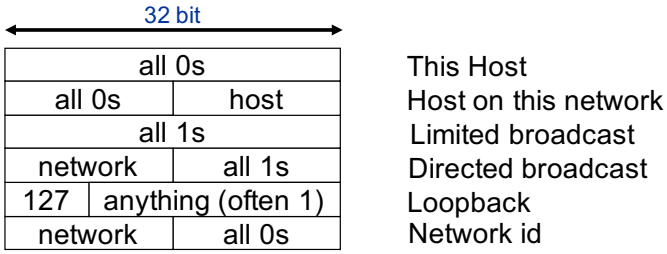
IP addr identifies **network** and **host** on that network
- Not machine but connection to network
- Device on n networks has n IP addrs – one for each

Address space administered by ICANN
- Assigned addrs don't have to be connected

IP Address Classes



Special IP Addresses



Addrs with all bits 0 or 1 are not assigned to hosts
- Useful at start-up if host/network not known

Broadcast is never valid source address

Loopback is for local inter-process communication (IPC)
- Should never exist on the network wire

Private Internet Address Ranges

Address ranges for internal use

10.0.0.0 - 10.255.255.255 (10/8 bit prefix)
172.16.0.0 - 172.31.255.255 (172.16/12 bit prefix)
192.168.0.0 - 192.168.255.255 (192.168/16 bit prefix)

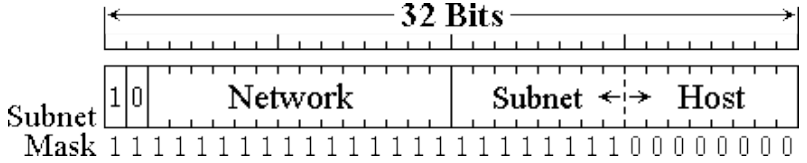
Addresses never routed on public Internet
- Not all devices need to be globally visible
- Used for testing and NAT (see later slides)

Subnets

As organisations grow, need finer control over network sizes
- Single class A/B/C network not good enough

Subnet is sub-network within assigned IP network
- To global Internet there is no distinction
- Internally subnet addrs may be used for routing, admin

Trade division into subnets for num of hosts in subnet
- Subnets can be any size within host field



Use high-order bits from host field to create subnets within network class:

subnet mask & address = network portion

Number of hosts and subnets

2^{subnet_bits} = number of subnets per network
- Although usage of all 0s and 1s is not RFC-compliant

$2^{(32 - network\_bits - subnet\_bits)} - 2$ = num of hosts per subnet
- All 0s and all 1s are not valid addresses

Subnet Example

In DoC, we have a class B network

- 8 bits for subnets and 8 bits for hosts
- Subnet mask 255.255.255.0 with class B net:
  256 subnets, each with 254 hosts

Example:

- 146.169.7.41 is global IP address of host 41 (columbia) on subnet 7 (DSE group) on IP network 146.169.0.0
- Full DNS name: columbia.doc.ic.ac.uk
- Broadcast to subnet on 146.169.7.255
- 7-net subnet mask of 255.255.255.0,
  DoC network mask of 255.255.0.0

Issues with IP Addressing

Support for mobility (laptops, phone, …)

- Connect to different points in different networks
- Routing depends on address used

Expansion of networks

- Renumbering/adding new number ranges hard
- Hosts with multiple IP addresses

Total size of address space limited

Address Space Problem

Shortage of unallocated addresses

- Practical address space in IPv4 is 100 million hosts
- IP is more popular than its designers expected

Some addr classes unnecessarily large

- Some organisations have more than they need
- Class B bigger than needs of most people
  - 64516 host addrs with 256 subnets of 254 hosts
  - Never mind class A!

Address Space Solutions

Stricter access to allocation

- Class A "virtually impossible" to obtain now
- Blocks of class C now allocated in preference to class B

Make address allocation more flexible

- Classless Inter-Domain Routing (**CIDR**)

Reuse addresses in different parts of network

- Network Address Translation (**NAT**)

Add more address bits: **IPv6**

Classless Inter-Domain Routing (CIDR)

Partition world into four zones

Allocate networks with variable subnet masks

- Size according to need, not just fixed classes A/B/C
- "Subnetting for global Internet"

Europe
194.0.0.0 - 195.255.255.255

North America
198.0.0.0 – 199.255.255.255

Central/South America
200.0.0.0 – 201.255.255.255

Asia & Pacific
202.0.0.0 – 203.255.255.255

Future use
204.0.0.0 – 223.255.255.255

Advantages

- More efficient allocation then previous classfull approach
- Works alongside previous allocations

Disadvantages

- Makes routing harder
- Not fundamentally larger address space

The Internet Map

courtesy of xkcd

Layout with a space-filling curve

"Old Internet" in top left corner

Go to caida.org for lots of pretty pictures

Network Address Translation (NAT)

Often only fraction of hosts require external access

Hide large network in small Internet address range

- External addr: real, allocated IP address
- Internal addrs: from private addresses range
- Gateway box translates internal addrs to dynamically allocated external addrs for traffic leaving LAN

Full address becomes IP addr + port *(next part)*

- External addr may be shared by multiple hosts over time
- Can lead to problems if changes aren't anticipated…

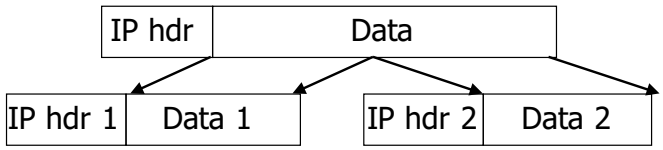
Fragmentation

Networks have maximum transfer unit (**MTU**)

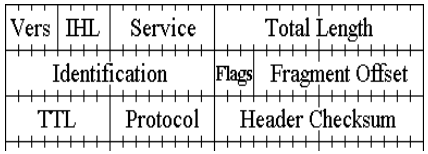
- LAN frames have limit on data carried
- e.g. Ethernet frame <1500 bytes
- Sender only knows datagram size for local network

Fragmentation of IP packets

- Large IP datagrams broken up when moving through network with smaller MTU
- Smaller datagrams need their own complete IP header



IP Datagram Fragmentation I



Fields to aid reassembling fragmented datagram

- Flags:
  - Bit 0: reserved, always 0
  - Bit 1: DF, 0 = may fragment, 1 = don't fragment
  - Bit 2: MF, 0 = last fragment, 1 = more fragments
- Fragment offset: position of fragment
  - In 8byte multiples, 1st is 0

IP Datagram Fragmentation II

All stations must accept fragments of <= 576 bytes

Final destination can reassemble original datagram

- Missing fragments are waited for
- Whole datagram discarded if any are lost
  - Best-effort connectionless delivery
  - Transport layer deals with missing datagrams

Fragmentation adds much complexity to routers

- e.g. multiple levels of fragmentation possible
- Often easier to return ICMP error message *(next slide)*

Internet Control Protocols

Address Resolution Protocol (ARP) - Mapping IP Addresses to Devices

DHCP

Internet Control Message Protocol (ICMP)

Mapping IP Addresses to Devices

Need to translate between addresses

- Data link layer: frames between devices use data link addrs, e.g. Ethernet MAC addrs
- Network layer: hosts send packets using IP addrs

Static mapping

- May be sufficient for small isolated network
- But Ethernet addr space is larger than IP addr space

But IP addresses are virtual

- No relation to hardware, maintained in software

IP supports interconnections of different networks

- Not all devices have Ethernet addresses

Dynamic Address Resolution

Need to bind protocol address dynamically
- Only possible for two devices on same network

Table lookup
- IP addr/data link addr in sequential/hash table

Closed-form computation
- Make physical addr simple function of IP addr

Message exchange
- Dedicated protocol for dynamic lookup, e.g. ARP
- Usual method on TCP/IP networks with static addresses, e.g. Ethernet

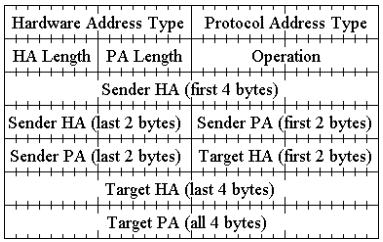
Address Resolution Protocol (ARP)

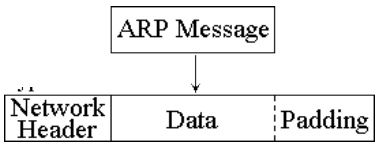
Hosts maintain caches of IP/data link address mappings for LAN

If host A has no entry for host B:
- A broadcasts **ARP request**
  - Requesting data link addr for B's IP address
- B recognises its IP address
  - Returns **ARP response** with its data link address
- B also caches A's data link/IP address mapping
  - Likely to need it in future exchanges

ARP is network layer protocol, not visible to the user

ARP Message Format





HW Addr Type: 1 = Ethernet

Proto Addr Type: 0800h = IP

HW Addr Length: 6 bytes

Proto Addr Length: 4 bytes

Operation: 1 = request, 2 = response

Target HW Addr: undefined on request

Target machine swaps target and sender in response

Reverse Address Resolution

Determine one's own IP address from Ethernet addr
- e.g. after booting machine

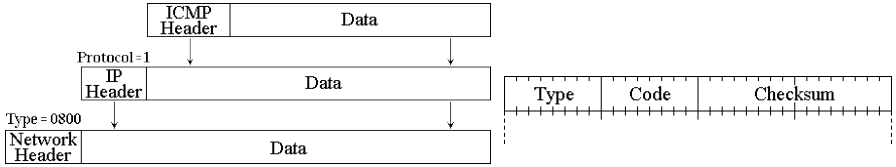
Use **RARP**, giving itself as both target and sender
- Need one RARP server per network
- Same format as ARP
- Limited broadcast of requests

Dynamic Host Configuration Protocol (DHCP)

Initially a host will only know its Ethernet address. DHCP used to assign dynamic IP addresses.

- Broadcast request for IP address (including source Eth addr) `DHCP Discover`
- DHCP server allocates address and issues `DHCP offer`
- Client (choses offer if necessary and) replies with `DHCP request`
- Address allocation confirmed with `DHCP ACK`
- Allocation for fixed period of time using leases. Host can request renewal or can release an address
- Also used to give client other configuration parameters, network mask, routers, host name, etc.
- Widely used for cable modems, WLANs, …

Internet Control Message Protocol (ICMP)



Allows routers to send control/error msgs to other routers/hosts

- Behaves as if higher level protocol, but integral to IP

ICMP provides for feedback about comms problems

- IP unreliable ➔ no guarantees of delivery, loss notification, control msg return

ICMP Message Format

Type (8bit) + code (8bit) gives kind of message

Type 3 codes:

0 = Net unreachable
1 = Host unreachable
2 = Protocol unreachable
3 = Port unreachable
4 = Fragmentation needed and DF set
5 = Source route failed

Checksum of type & code, 1s compliment

Some Types:

0 = Echo reply
3 = Destination Unreachable
5 = Redirect
8 = Echo request (ping)
11 = Time exceeded
12 = Parameter problem
13 = Timestamp
14 = Timestamp reply
15 = Information request
16 = Information reply
17 = Address mask request
18 = Address mask reply

Popular client uses of ICMP

Ping

- Used to verify that path works and end host present.
- Collects round trip time and failure
- Send echos, display echo reply.

Traceroute

- How to find out info about intermediate hosts?
- Send packets and increment TTL at each packet.
- When TTL=0 router discards packet but sends ICMP error message.

TRY Them!!

IPv6

IETF addresses many problems of IPv4 with **IPv6**

128 bit addresses (vs. 32 bit in IPv4)

- $3.4*10^{38}$ unique host addresses (vs. $4.2*10^{9}$ in IPv4)

Simplified 7 field header (vs. 13 fields in IPv4)

- Faster processing in routers possible
- More options through extension headers
- Support for authentication, privacy, service types, mobility, …

Compatible with IPv4 for transition

- Some gateways and tricks to hide IPv6's greater capabilities

Issues with IPv6

Difficult to implement properly

Transition from IPv4 to IPv6 hard and slow

- Not widely deployed over backbone
  - Router/switch manufacturers not pushing it
  - ISPs and network providers not demanding it
- Many of the benefits lost in gateways to IPv4

Currently useful within organisation

- But not many people to talk IPv6 with
- Mobile phones may push adoption…