Internet Protocol - IP

IP

IP addressing and subnetting

IP Encapsulate, Format and package

Internet protocol (IP)

- Major layer 3 protocol
  - While layer 2 is concerned with data transmission from one device to a directly adjacent device, layer 3 is concerned with delivery of data across the internet, from once arbitrary device to another

- Provides services to Layer 4. Takes data packaged by TCP and UDP (layer 4 protocols and delivers it across the internet
- Independent of underlying physical network (could be Ethernet, wireless, PPP, etc)
  - Layer 1 defines the ***bit***, so that we can digitise
  - Layer 2 defines the ***frame***, which is technology dependent

IP cont

- A ***connectionless protocol***. No connection between end points A and B is set up
- ***Delivered unreliably*** – “best effort” protocol
- ***Delivery without acknowledgement***

- Such features are optionally) provided by higher level (layer 4) protocols

- Also provided by some *but not all* layer 2 protocols (eg WiFi)

IP Functions

- Addressing
- Encapsulation, Formatting and packaging
- Fragmentation and Reassembly
- Routing / Indirect Delivery

IP addressing

- 32 bit binary addresses
- Difficult for humans to read so commonly represented as “dotted decimal” – four decimal numbers in the range 0-225, representing each byte, separated by “.”
- Eg 25.176.0.1   (hex 19.B0.0.1)

IP addressing schemes

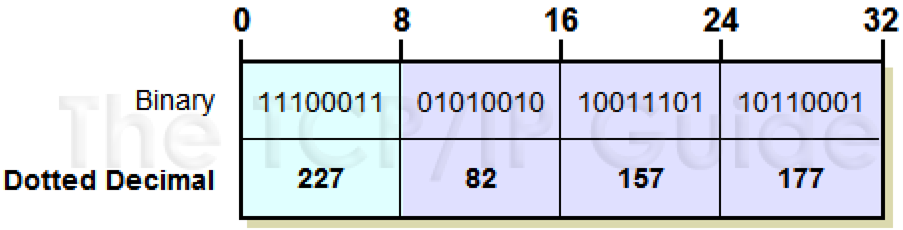
- An IP address consists of a network component, possibly a sub net component, and a host component.
- Different ways of dividing up the IP address space have been developed as the need to have more and more IP addresses has arisen.
- The first was called *classful*, then *sub-netting* was developed, followed by *classless*

IP addressing schemes

- The original intent of IP addresses was that they be unique across the internet, however this is no longer possible for the standard 32 bit IP addresses.
- The continued need for more IP addresses has lead to the development of a whole new TCP/IP standard called IPv6
- However this is not widely used. The current standard (IPv4) has been extended by means of IP NAT and port addressing (discussed later)

IP addresses

- IP addresses consist of an network component and a host component:
- In the example below, 8 bits are allocated to the network id, and 24 to the host id:



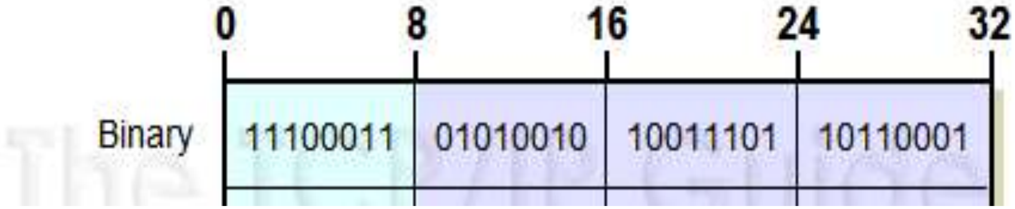
	0	8	16	24
Binary	11100011	01010010	10011101	10110001
Dotted Decimal	**227**	**82**	**157**	**177**

IP Address: 227.82.157.177
Split Into 8-Bit Network ID and 24-Bit Host ID

IP classful addressing

IP Address Class	Fraction of Total IP Address Space	Number Of Network ID Bits	Number Of Host ID Bits	Intended Use
Class A	1/2	8	24	**Unicast addressing for very large organizations with hundreds of thousands or millions of hosts to connect to the Internet.**
Class B	1/4	16	16	**Unicast addressing for medium-to-large organizations with many hundreds to thousands of hosts to connect to the Internet.**
Class C	1/8	24	8	**Unicast addressing for smaller organizations with no more than about 250 hosts to connect to the Internet.**
Class D	1/16	n/a	n/a	**IP multicasting.**
Class E	1/16	n/a	n/a	**Reserved for “experimental use”.**

- Because of the way the address space is divided up, the address indicates the class



0	8	16	24	32

Binary	11100011	01010010	10011101	10110001

- If bit 0 = 0, then Class A
  - else if bit 1 = 0 then Class B
    - else if bit 2 = 0 then Class C
      - else if bit 3 = 0 then Class D

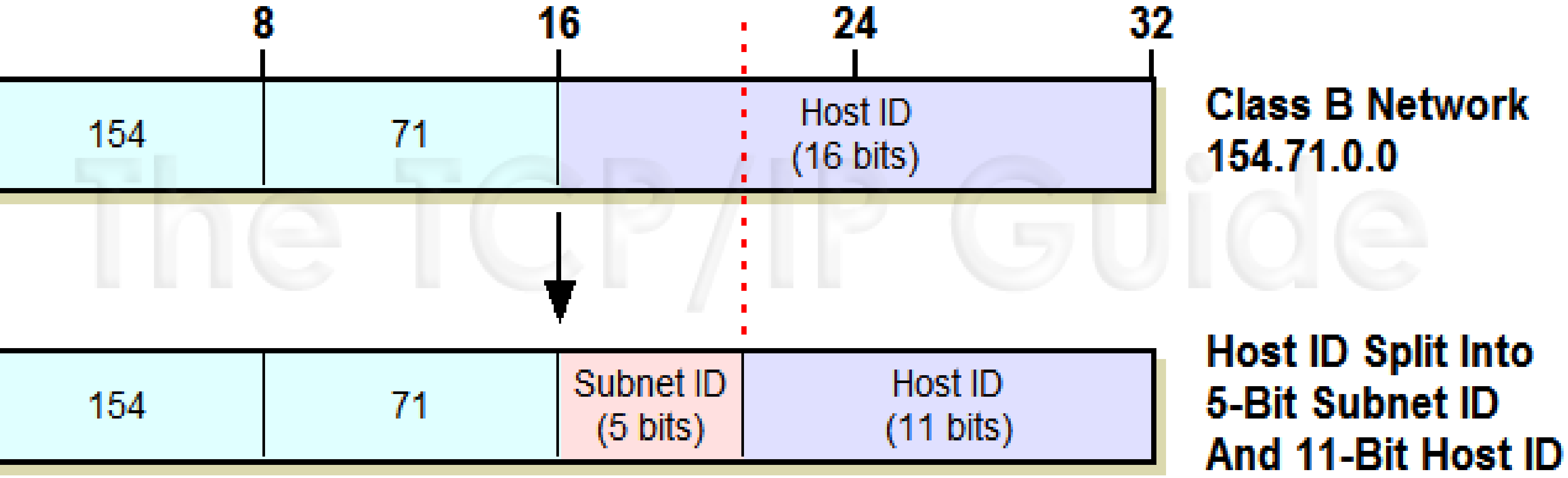
else Class E

Subnetting

- *While the class based addressing was ok, problems arose as the internet expanded*
- Organizations don’t have large numbers of devices connected to the one huge network (the would be inefficient)
- Sub-netting was developed to address this
  - It supports the concept of an organization having a number of smaller networks (LANs) that are interconnected

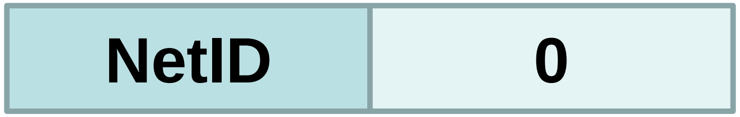
Subnetmask

- Subnetting takes the host component of an address end splits it onto a subnet id and a host id
- In other words, a subnet behaves the same as a net, except that it works on the local part of the net address.



Netmask

- The network address is the NetID part, with the HostID part set to all 0's.
- A hostID of 0 is the address of the router responsible for this network.

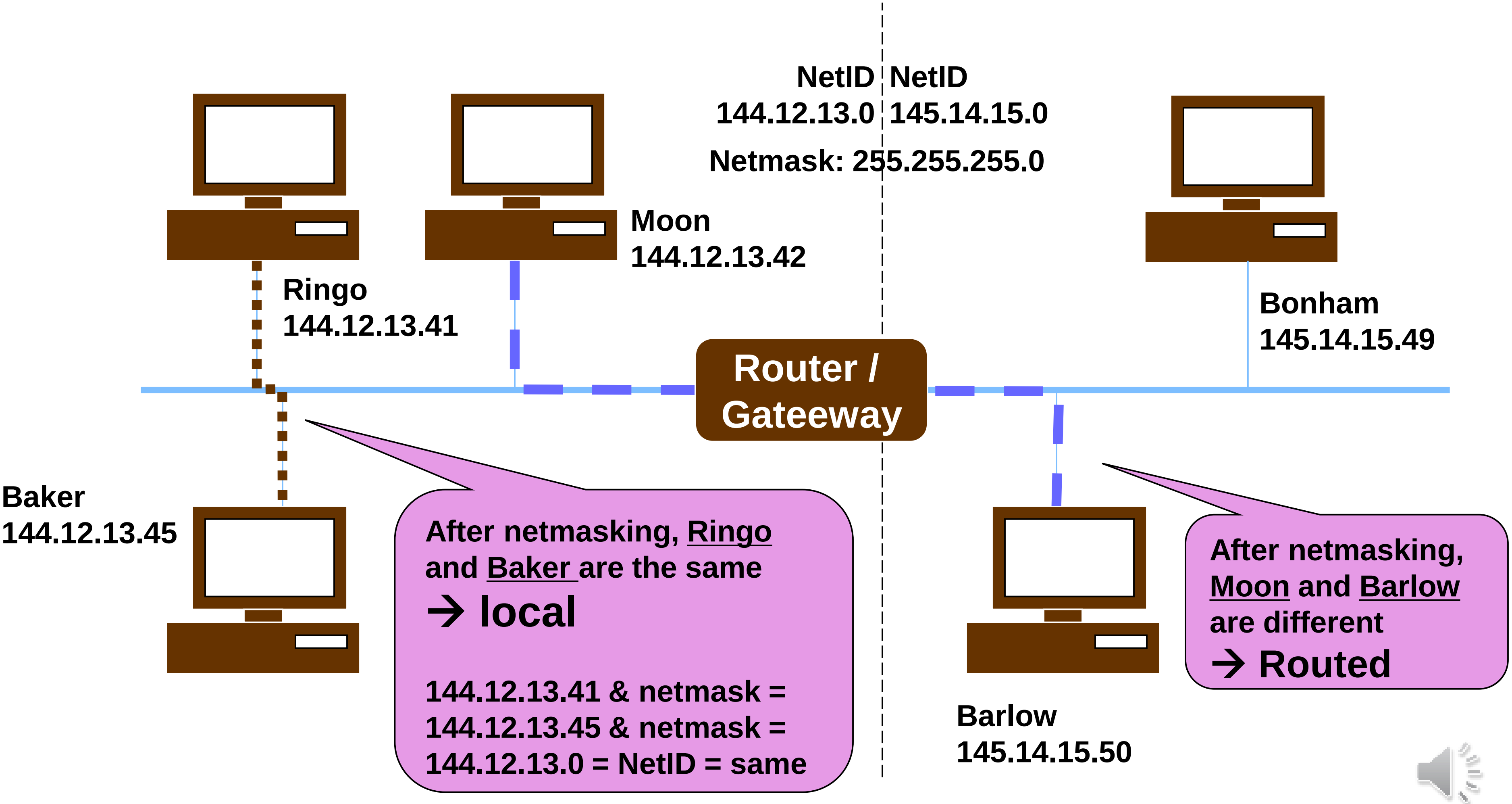


- Applying the netmask:

```
131.170.5.125 = 10000011 10101010 00000101 11111101
255.255.0.0   = 11111111 11111111 00000000 00000000
AND
131.170.0.0   = 10000011 10101010 00000000 00000000
```

Using the Netmask

- The netmask is used to determine whether the transmission is local or through the router. This is done by comparing the source/dest address after AND masking by the netmask,



NetID 144.12.13.0

NetID 145.14.15.0

Netmask: 255.255.255.0

Ringo 144.12.13.41

Moon 144.12.13.42

Bonham 145.14.15.49

Router / Gateway

Baker 144.12.13.45

Barlow 145.14.15.50

After netmasking, Ringo and Baker are the same

→ local

144.12.13.41 & netmask =
144.12.13.45 & netmask =
144.12.13.0 = NetID = same

After netmasking, Moon and Barlow are different

→ Routed

CIDR: Classless Inter-Domain Routing

- *Subnetting helped, but was still not enough to handle the increase in IP addresses. The solution was to do away with classes altogether.*

- The CIDR notation is used to specify where the boundary between the network prefix and the host suffix occurs and the notation is A.B.C.D / N, but unlike the traditional classes, it is much more flexible. The NetID can now be specified in ***bits***.

- For example, the network address 131.170.0.0/16 specifies that the first 16 bits (two bytes) of the IP address identifies the network.
  - The remainder of the IP address, in this case the last two bytes, or 16 bits, can take any value, and the complete IP address represents an actual host on the network
  - This is the same as a Class B network, but using CIDR, this could be shifted by 1 bit. If the IP network address becomes 131.170.0.0/17, then the size of the network is larger. The protocol specifies that the host suffix of the IP address cannot use all zeros or all ones.
    - Why? These are special addresses (see later)

The CIDR Netmask

- So a CIDR netmask is also a bit mask (1’s and 0’s) that specifies how many bits make up the address prefix.
- The netmask for the address prefix or NetID

  131.170.192.0/19

  is

  1111111.1111111.**111**00000.00000000.

- That is, where there is a 1 in the netmask, that portion of the IP address is part of the **address prefix** (this is done with a logical AND operation).

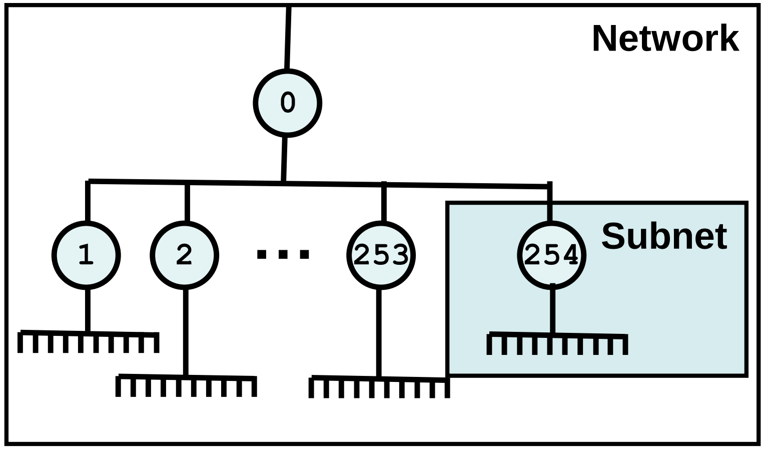
- This IP address using a netmask would look like

  131.170.0.0/255.255.224.0.   (224 = 128+64+32)

Subnets

- All our discussions abou net and netmasks also apply to subnets.
  - A subnet is simply a network within the organisational network.

- Advantage is that a SubnetID
  - can be changed as a whole
  - can be secured as a whole
  - can be hidden as a whole



Subnet – Example 1

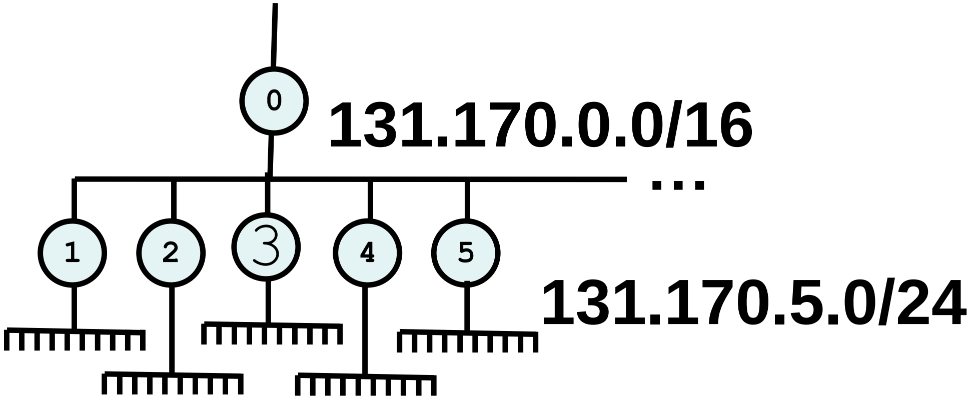
Org NetID **131.170.0.0/16, HostID = 16 bits**

Org Netmask **11111111 11111111 00000000 00000000**

Suppose ~ 255 subnets desired, NetID extended by 8 bits (2^8=256)

SubNetID **131.170.192.0/24, HostID = 8 bits (smaller)**

SubNetmask **11111111 11111111 11111111 00000000**



131.170.0.0/16

131.170.5.0/24

Subnet – Example 2

- Net = 141.123.0.0/13
- Netmask = 11111111 11111000 00000000 00000000
- 11-bit Subnets required
- Subnetmask = 11111111 11111111 11111111 00000000

- So there are now 2^{8}-2 local addresses in a subnet, and 2^{11}-2 addresses for the whole orgamisatopm/

IPv4 addresses with special functions

0.0.0.0 / 8

- often used to read from 'any' NIC
  in a router 0.0.0.0/8 **= *read from ANY***

0.0.0.* / *

- rarely used, 0.0.0.0/* **= *write to Local 1***

***.*.*.0 / 32**

- where netmask has HostID = 0, * = network
  address *.*.*.0/32 = ***write to router/gateway***

255.255.255.255 / 32

- 255.255.255.255 /32 = ***write to ALL***

127.0.0.1 / 8

- 127.0.0.1/8 = ***write to self***

224.0.0.0 / 4 to 239.255.255.255 / 4

- used for **multicast** networks

10.*.*.* / 8,   172.16.*.* / 12,   192.168.*.* / 16

- Private **Unroutable** Addresses (*will NOT be routed anywhere*)

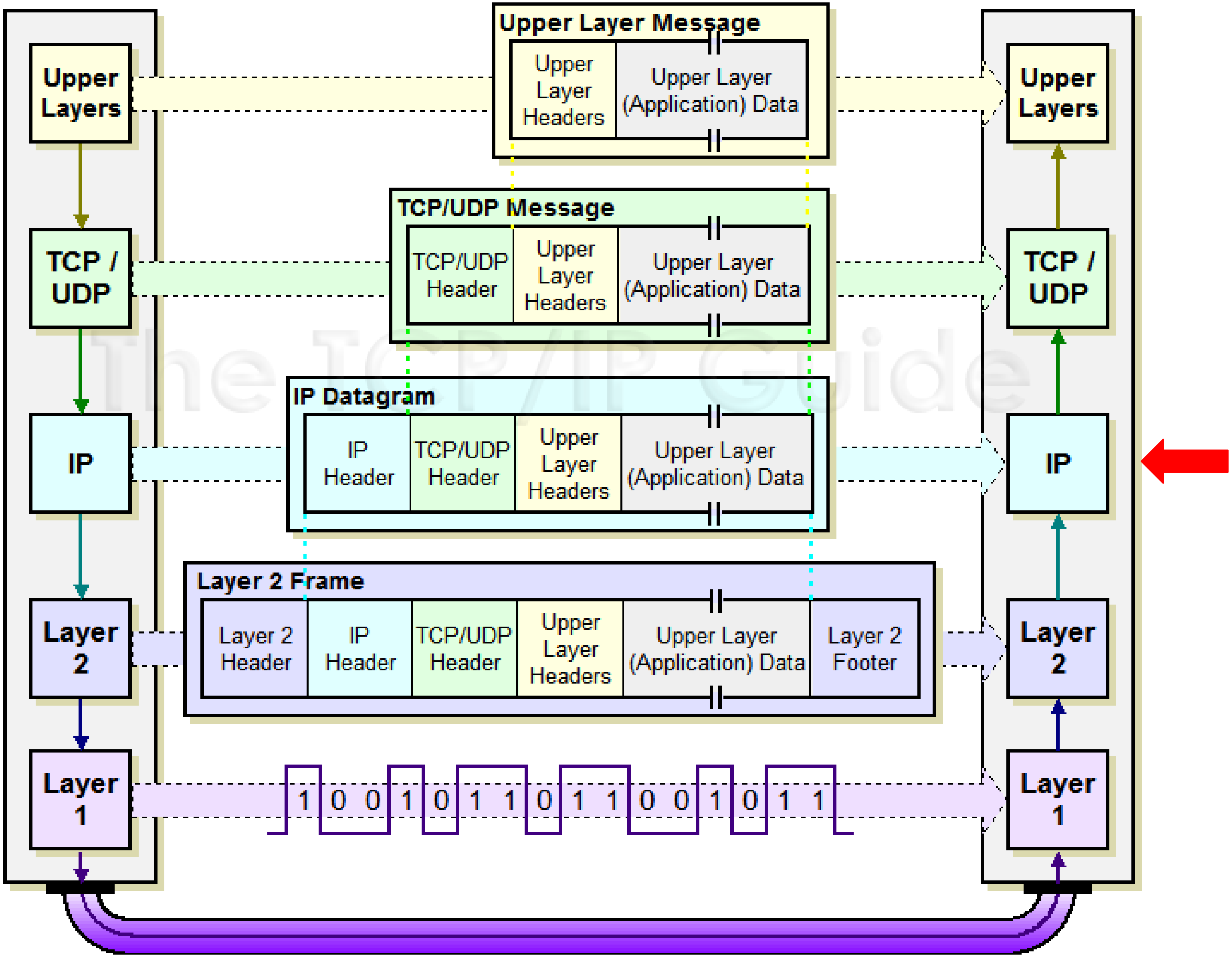
0	**0**	**SELF**
0	**HostID**	**LOCAL**
NetID	**0**	**NET SELF**
NetID	**1**	**BROADCAST**
1	**1**	
127	**1**	**LOOPBACK**
224 – 239	**HostID**	**MULTICAST**
10,176,192	**HostID**	**UNROUTED**

IP Encapsulation, Formatting and packaging

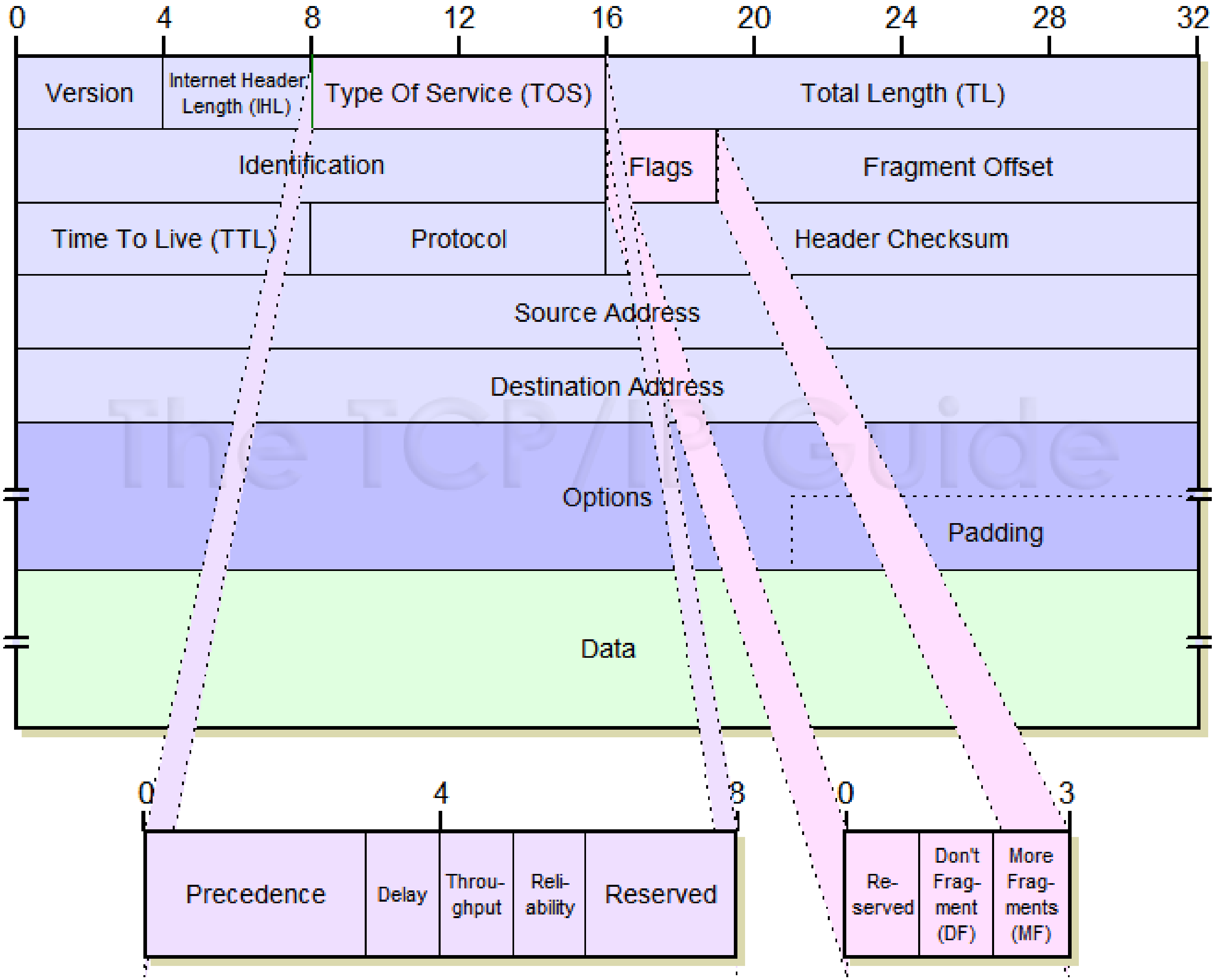
- The primary job of the Network layer is to deliver data over the internet, possibly travelling across many physical networks.
- Takes data from higher levels and packages it for transmission.
- The unit of transmission is called a ***datagram***. Higher layer messages may need to be broken into several *datagrams*

- Independent of the type of network used for transmission (often several different types of physical networks)

Layer 3 – IP


Upper Layer Message
Upper Layer Headers
Upper Layer (Application) Data
TCP/UDP Message
TCP/UDP Header
Upper Layer Headers
Upper Layer (Application) Data
IP Datagram
IP Header
TCP/UDP Header
Upper Layer Headers
Upper Layer (Application) Data
Layer 2 Frame
Layer 2 Header
IP Header
TCP/UDP Header
Upper Layer Headers
Upper Layer (Application) Data
Layer 2 Footer
Upper Layers
TCP / UDP
IP
Layer 2
Layer 1
Upper Layers
TCP / UDP
IP
Layer 2
Layer 1
1 0 0 1 0 1 1 0 1 1 0 0 1 0 1 1
The TCP/IP Guide

Datagram format (IPv4)



Datagram format (IPv4)

- A datagram consists of a *header* and a *payload*.
- The header includes
  - Version
  - Internet Header length
  - Type of service
  - Total Length
  - Identification
  - Flags
  - Fragment offset
  - Time to live Protocol
  - Header Checksum
  - Source Address
  - Destination Address
  - Options
  - Padding

Datagram format (IPv4)

- ***Fragment offset***
  - the offset of this fragment in the original message (13 bits)

- ***TTL***
  - Time to live decremented with each router hop – designed to detect infinite loops in network routing

- ***Protocol***
  - Identifies the higher level protocol carried in the datagram

Datagram format (IPv4)

- ***Checksum***
  - 16 bit CRC
- ***Source and Destination Address***
  - 32 bit IP addresses. The destination address is the ultinate address, not the next router
- ***Options***
  - Various additional things include the option to record or specify the route to be taken, timestamps and data used for traceroute