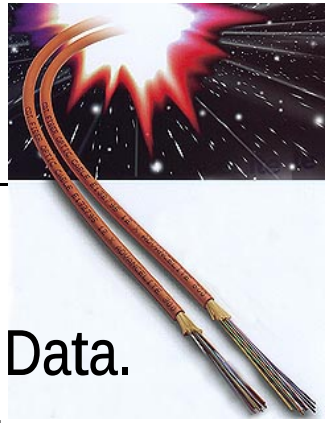
Data Agenda

- Some basic data concepts.
- Modems
  - Dial up
  - Digital Subscriber Line
- Fiber to the home
- Hybrid fiber/coax
- Ethernet
- Networks and Protocols – Introduction
- Frame Relay
- Asynchronous Transfer Mode (ATM)
- The Internet – TCP/IP
- Multiprotocol Label Switching (MPLS)
- Optical Transport Network (OTN)

Introduction



- This presentation is divided into two parts: Voice and Data.
- Originally, the network was essentially all voice and when data usage began to grow, the first efforts to communicate data were directed at using the existing circuit switched network.
- Eventually, the network became majority data, to the point that voice is now carried on the data network.
- I'll start with how voice is carried in the traditional circuit switched network, and then show the transition to a data centric network. I will not cover how the old analog telephone network operated because that's now obsolete.

Some Basic Concepts

Open Systems Interconnection (OSI) Reference Model

- The OSI was developed by the International Organization for Standardization (ISO), headquartered in Geneva, Switzerland.
- The reference model they developed was intended to be a way to divide communication software systems so that they had well defined interfaces.
- While not always honored completely, it gives us a way to talk about a communication system.

OSI Reference Model

OSI Reference Model

7 – Application Interface to end user. Interaction directly with software application.		**Software App Layer** Directory services, email, network management, file transfer, web pages, database access.	FTP, HTTP, WWW, SMTP, TELNET, DNS, TFTP, NFS
6 – Presentation Formats data to be "presented" between application-layer entities.		**Syntax/Semantics Layer** Data translation, compression, encryption/decryption, formatting.	ASCII, JPEG, MPEG, GIF, MIDI
5 – Session Manages connections between local and remote application.		**Application Session Management** Session establishment/teardown, file transfer checkpoints, interactive login.	SQL, RPC, NFS
4 – Transport Ensures integrity of data transmission.	Segment	**End-to-End Transport Services** Data segmentation, reliability, multiplexing, connection-oriented, flow control, sequencing, error checking.	TCP, UDP, SPX, AppleTalk
3 – Network Determines how data gets from one host to another.	Packet	**Routing** Packets, subnetting, logical IP addressing, path determination, connectionless.	IP, IPX, ICMP, ARP, PING, Traceroute
2 – Data Link Defines format of data on the network.	Frame	**Switching** Frame traffic control, CRC error checking, encapsulates packets, MAC addresses.	Switches, Bridges, Frames, PPP/SLIP, Ethernet
1 – Physical Transmits raw bit stream over physical medium.	Bits	**Cabling/Network Interface** Manages physical connections, interpretation of bit stream into electrical signals	Binary transmission, bit rates, voltage levels, Hubs

OSI Reference Model

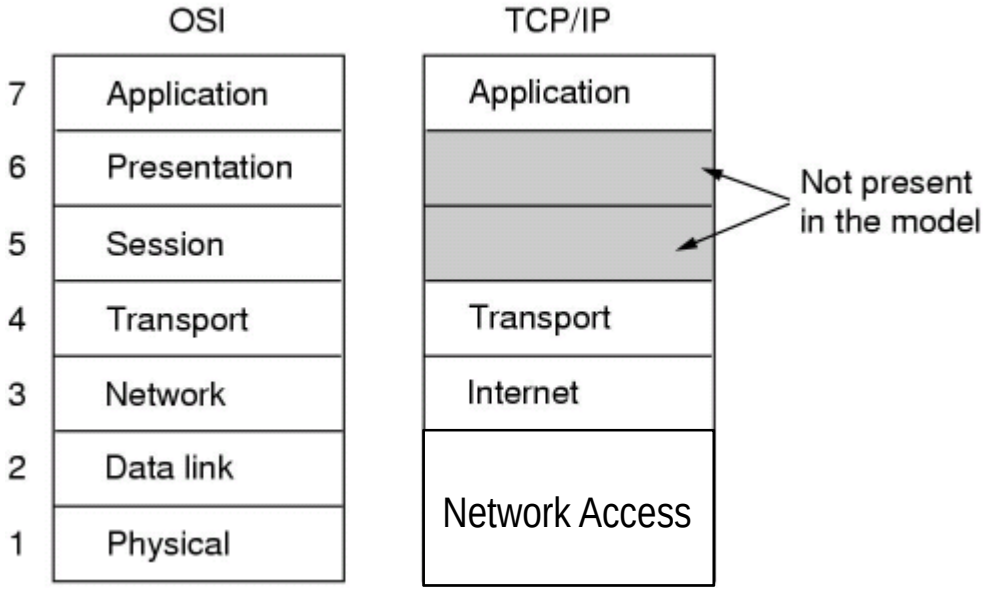
- The OSI reference model consists of seven levels:
  - Layer 1 – the Physical Layer. This layer consists of the physical medium used for transmission – fiber, copper, wireless, etc.
  - Layer 2 – The Data Link Layer. This layer transmits and receives frames of data and recognizes link addresses (such as Ethernet MAC addresses). Checks for errors, if function is included in the protocol.
  - Layer 3 – The Network Layer. Responsible for establishing a connection from station to station across an internetwork. IP is in this layer.
  - Layer 4 – The Transport Layer. Provides reliable end-to-end error recovery mechanism. TCP is in this layer.

OSI Reference Model

- Layer 5 – The Session Layer. Provides a mechanism for establishing reliable communications between applications on separate computers.
- Layer 6 – The Presentation Layer. Provides a mechanism for dealing with data representations in applications.
- Layer 7 – The Application Layer. The end-to-end application. Email and the WWW are examples.

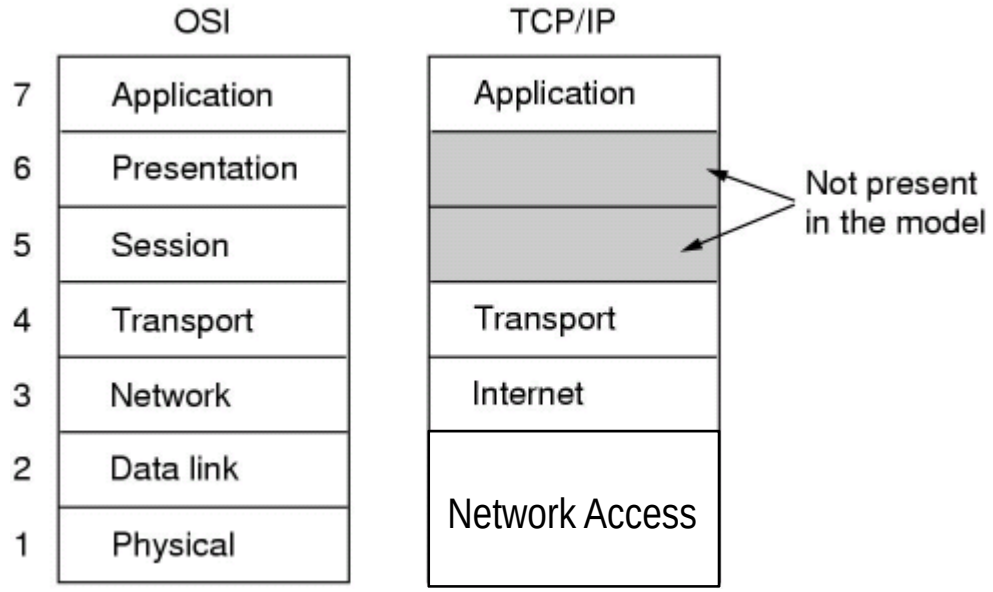
An Alternate Reference Model (Internet)

- The OSI model was “prescriptive”. That is, it was developed as a model of how to develop software layers.
- The Internet, however, developed without reference to the OSI model and when the software was completed, the designers described the layers they had developed. The Internet model is “descriptive”.



Internet Layers

- In the Internet, the lowest layer combines the first two layers of the OSI model under the name “Network Access”.
- This can cause problems when someone speaks of, for example, Layer 3 functions. You need to know which model they’re using.
- We’ll talk about TCP/IP a lot more later in this presentation.



Circuit and Packet

- In the voice portion of this presentation, we mostly focused on the concept of a “circuit” and “circuit switching”.
- In this data portion, I’ll introduce the concept of “packet” and “packet switching (or routing)”.
- The concept of a circuit still exists in the packet world, as “virtual circuits,” usually where all packets follow the same path through the network.

Frames and Packets

- Technically, a frame is a block of data which is delineated in some way, such that you can find the start of the frame.
  - For all the frames we talk about, I'll describe how the system finds the start of the frame.
- A packet may or may not have a way to delineate the start of the packet.
  - In common usage, people often interchange the terms frame and packet.

Connection and Connectionless

- When we want to communicate data between two end stations, there are two paradigms we can use: Connection and connectionless.
- With a connection, we establish a logical path between the two end stations prior to sending the data. Theoretically, the called station could reject the connection request.
- With connectionless, we put the address of the other end station on the data and send it into the network. The network has the responsibility of delivering the data based on the address. Something like putting a letter into a mail box.

  Each packet is treated independently so packets may take different routes and/or arrive out of order.

Introduction to Data Transmission

- The need to send data over the telephone network started early.
- In 1958, AT&T released the Bell 101 modem, which had been developed for the SAGE System. It communicated at 110bps.
- In 1962, they followed with the Bell 103 modem which communicated at 300bps.
- By 1972, rates had increased to 1200bps.
- Improvements in the network (digital network) allowed better modulations and faster speeds.
- The V.34 standard in 1994 was the last “pre-Internet” modem.

Introduction to Data Transmission

- The Internet took off in about 1996 and that created an explosion of innovation in data communications.
- We'll cover some of that technology in this section.

Challenges of Data Communications

- Need to communicate at high rates, yet be reliable.
  - In 1980, we thought a 1200bps telephone line modem was fast.
  - Today, we have 100Mbps over wireless (a much more difficult medium).
- Sharing of transmission channels.
  - Impossible to give everyone a dedicated path.
- Need to scale the networks.
  - Things that work in small systems get very difficult in large systems.

Modems

- Modem is a contraction of modulator/demodulator.
- When speaking about modems we talk about bits per second (bps) and baud (symbols per second).
- Many people incorrectly equate baud with bps, so I'll often use the term "symbols" instead of baud.
- The term "baud" is named for Jean-Maurice-Émile Baudot who invented the baudot code.

Early Modems

- In the very early days, users could not connect non-AT&T modems directly to the telephone network because of legal restrictions, so an acoustic coupler had to be used.
- This also implied that tones had to be used to carry the data, one tone for a 1 and another tone for a 0.
- These modems used a modulation called Frequency Shift Keying (FSK).

Early Modems

- These early dial-up modems were used with “dumb terminals”, essentially CRT versions of teletypewriter (teletype) terminals (TTY).
- The communication was asynchronous because that’s what TTY terminals were.
- Often used to access a remote timesharing computer.
- AT&T offered modems that could be connected directly to the line, both for dial-up and for use on leased lines.
  - Most leased lines were 4-wire, two for transmit and two for receive, so the full bandwidth could be used to transmit data.

FSK Modulation

- I'll use the Bell 103 (300 bps) modem as an example.
- The Bell 103 is a full duplex modem, using tones a follows:

	Originating modem	Answering modem
Mark (1 bit)	1,270Hz	2,225Hz
Space (0 bit)	1,070Hz	2,025Hz

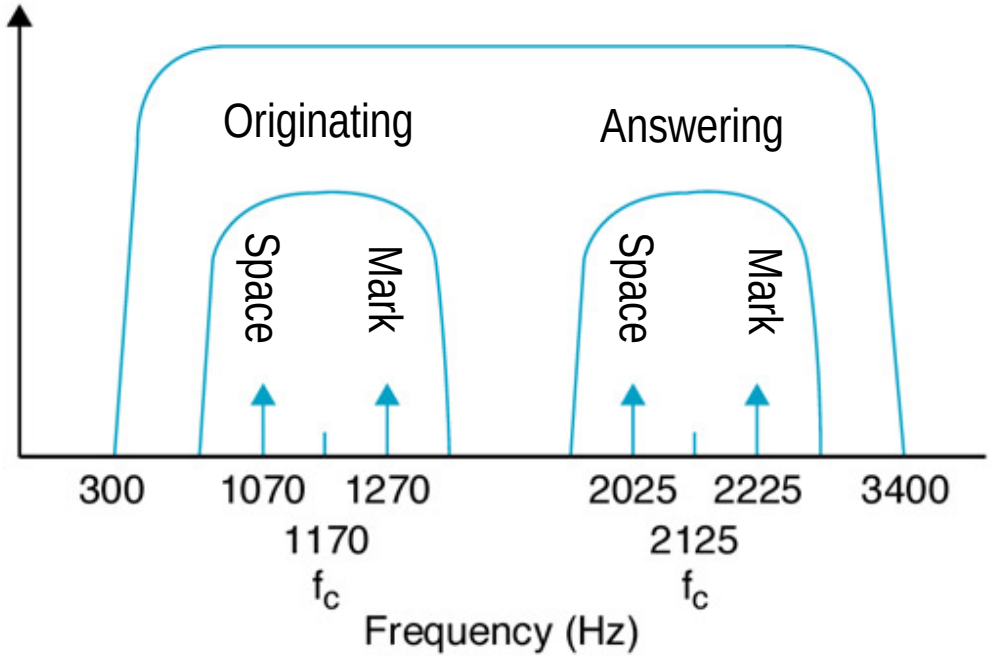
- Each modem had filters for the frequencies they were receiving, so they could detect whether a 1 or a 0 was sent.
- Transmission was slow because the frequency had to be sent long enough for the receiver to discriminate which tone was transmitted.
- Note that these frequencies fit within the 300Hz to 3400Hz bandwidth of the telephone channel.

FSK Modulation

- Let's look at the frequency spectrum of the Bell 103 modem compared to the bandwidth of a voice channel.
- Since a 1 and 0 is never sent simultaneously, the bandwidth of the mark and space can overlap – but not with the other modem.

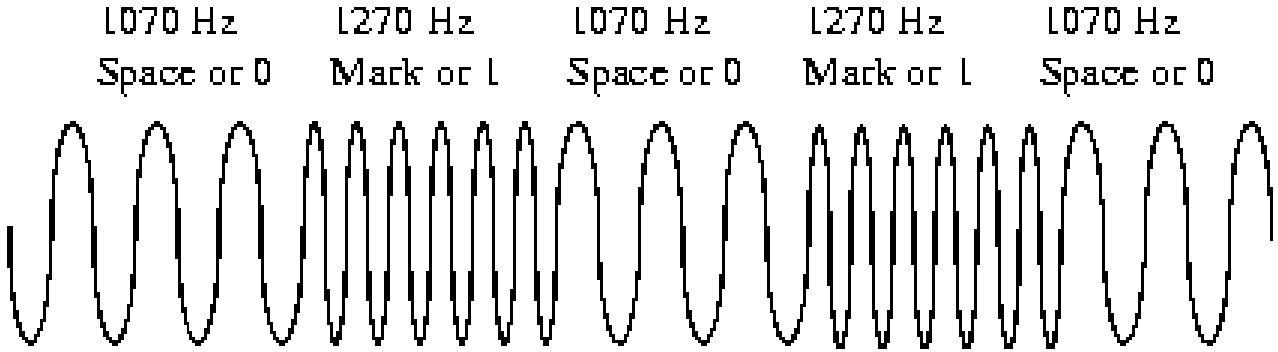
The bandwidth of each frequency is about 600Hz (300Hz on either side of the center frequency) so the mark of the originating modem goes from about 970Hz to 1570Hz.

The space bandwidth goes from about 770Hz to 1370Hz.



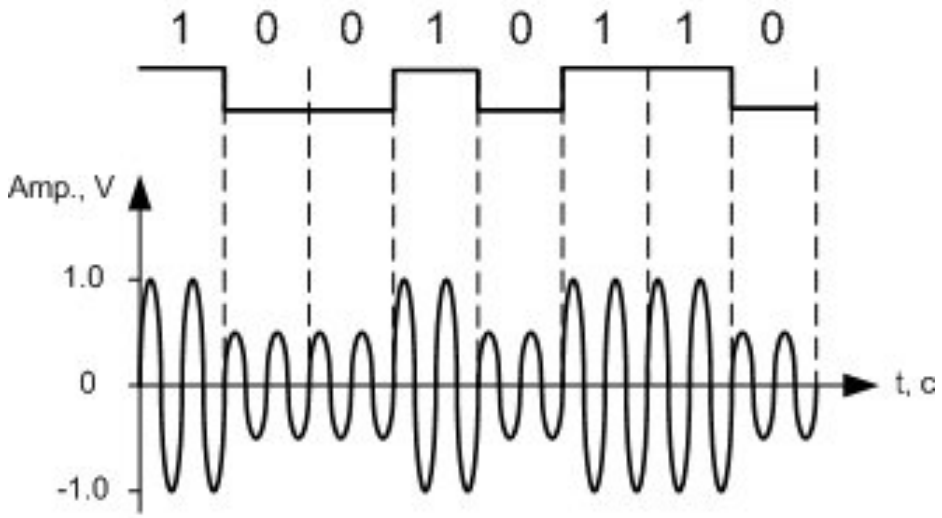
FSK Modulation

- Here's what the transmit signal of the originate modem would look like when transmitting the sequence {0,1,0,1,0}.
- For the originating modem, the center frequency is 1170Hz, and the delta frequency ($\boldsymbol{\Delta F}$) is 100Hz. The total bandwidth of FSK is 2*$\boldsymbol{\Delta F}$ + 2*bit rate, or 200+600=800Hz
- FSK is not very bandwidth efficient – significantly less than 1 bit per Hz (3/8 bit per Hz in this example).
- But it is tolerant of noise because noise does not affect the frequency.



Amplitude Shift Keying (ASK)

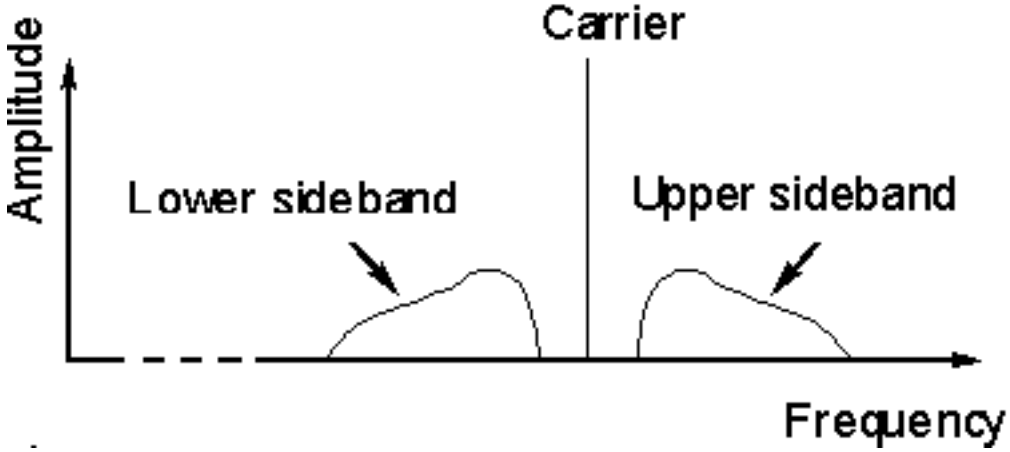
- Although there were no early modems that used ASK, it's important to understand it because the concept is used in Quadrature Amplitude Modulation (QAM).
- In ASK, the amplitude of the carrier signal is varied between two levels.
- Although there's no standard, the lower amplitude was often a 0 and the higher amplitude was a 1.



Amplitude shift keying (ASK)

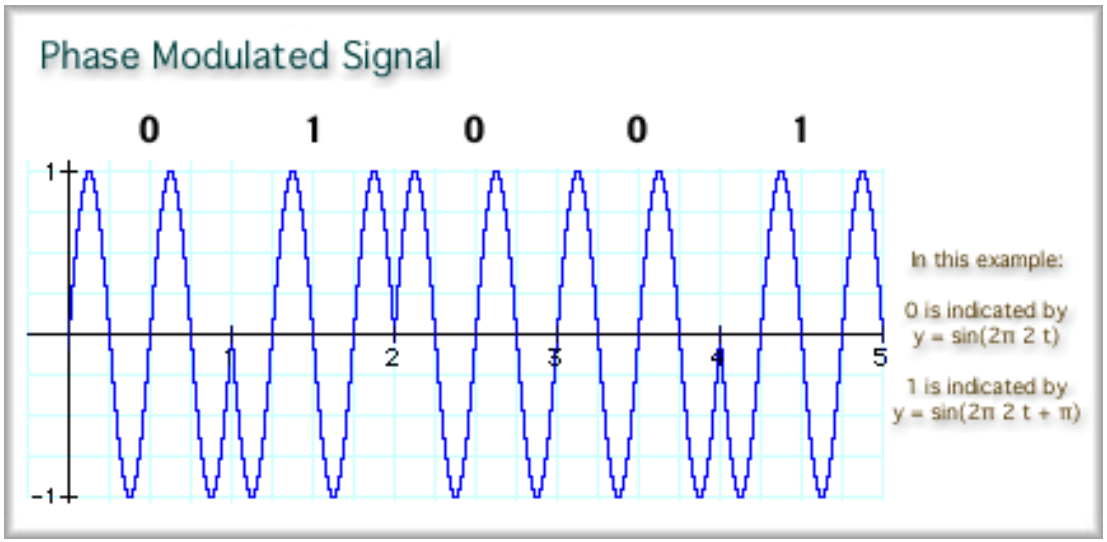
Amplitude Shift Keying (ASK)

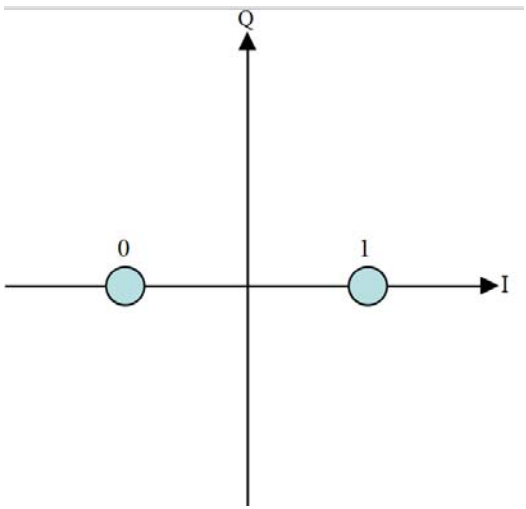
- ASK is plain old amplitude modulation so the signal will have two sidebands, each sideband extending to the maximum frequency of the modulating signal.
- For data, a Hz is a combination of a 0 and a 1, so the maximum modulating frequency is half the bit rate, meaning that the bandwidth is equal to the bit rate.
- The average modulating frequency will be less than the maximum because the data will not be all alternating 1s and 0s.



Phase Shift Keying (PSK)

- In phase shift keying, the phase of a carrier wave is varied to indicate the digital data.
- For Binary Phase Shift Keying (BPSK), two phases are used, one representing a 1 and the other representing a 0.
- The bandwidth of the signal is equal to the symbol rate, which for BPSK is the same as the bit rate.

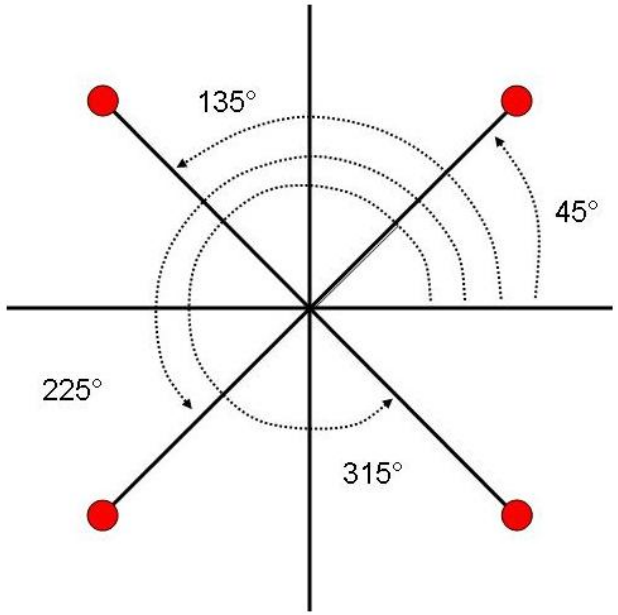




Phase Shift Keying (PSK)

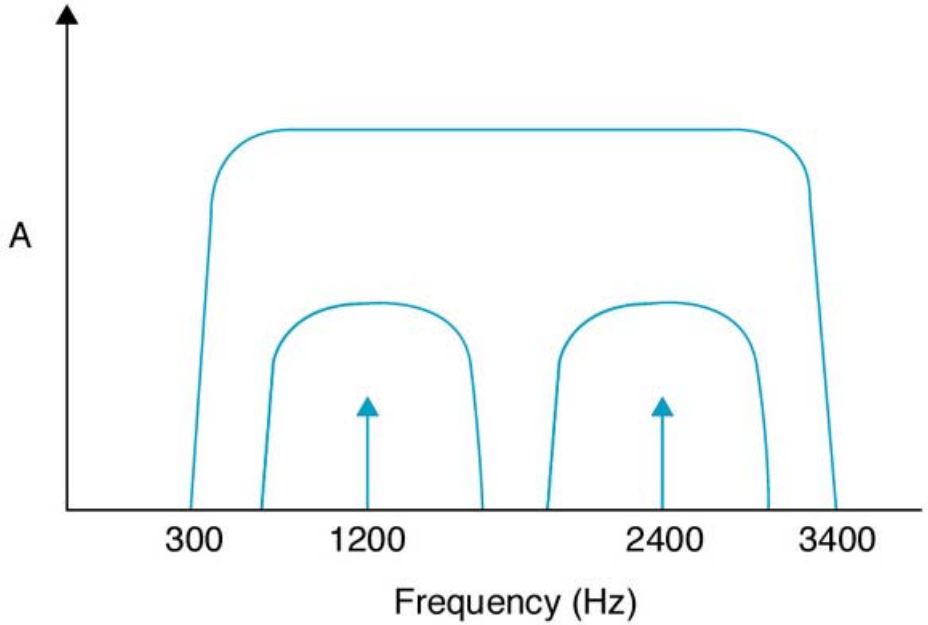
- More than two phases can be used. If four phases are used, each phase can carry 2 bits.
- The constellation diagram below shows a four phase signal, using the phases 45º, 135º, 225º and 315º.
- Eight phase shifts would allow each phase to carry 3 bits.
- General case is 2^n phase shifts can each carry n bits.

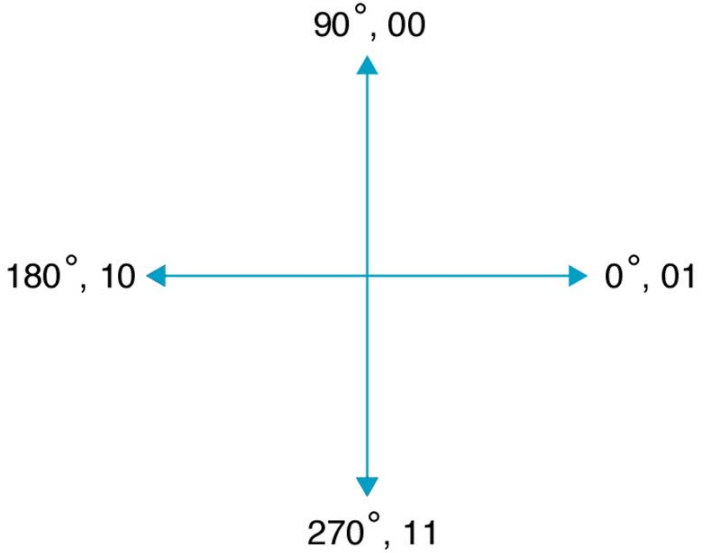
The Bell 212A and V.22 modems used PSK at 600 baud to provide 600bps and 1200bps.



Phase Shift Keying

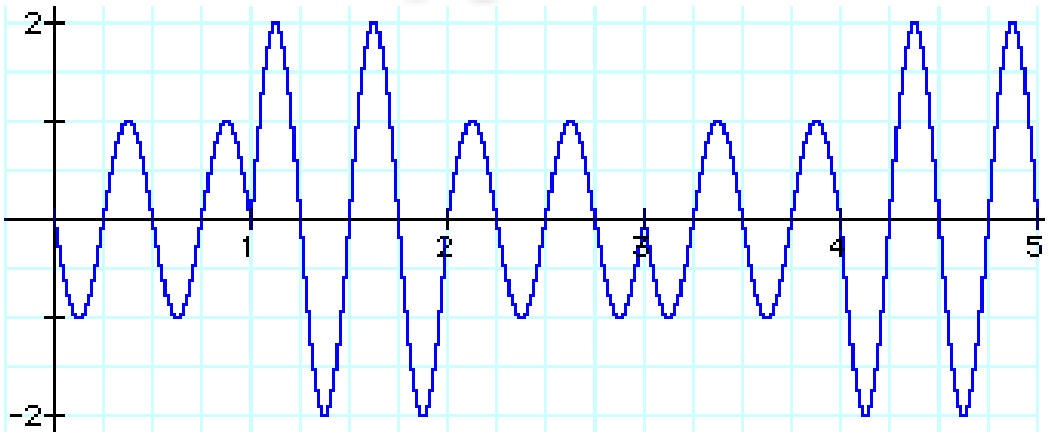
- An example of a PSK modem is the Bell 212A modem (and ITU V.22).
- It operated 1200bps full-duplex with two carriers located at 1200Hz and 2400Hz. Symbol rate and bandwidth of each was 600Hz.
- Each symbol carried 2 bits for 1200bps operation.





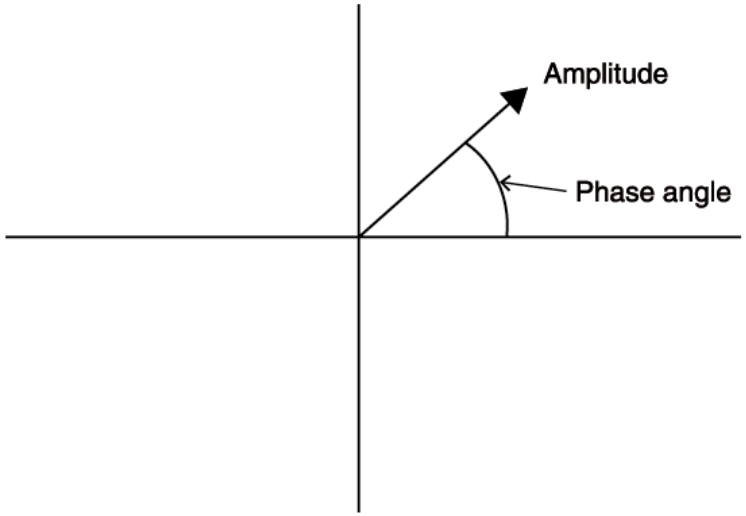
Quadrature Amplitude Modulation (QAM)

- While PSK is an efficient modulation, allowing more than one bit per Hz of bandwidth, it has limits.
  - As the number of phases increases it becomes harder for the receiver to accurately decode the data. The phase changes are just too small.
- If we add amplitude variations to each phase, we can create more symbols that can be accurately received.

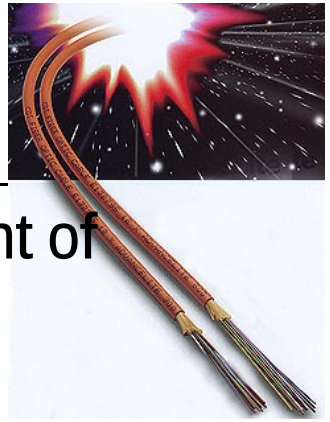


Quadrature Amplitude Modulation (QAM)

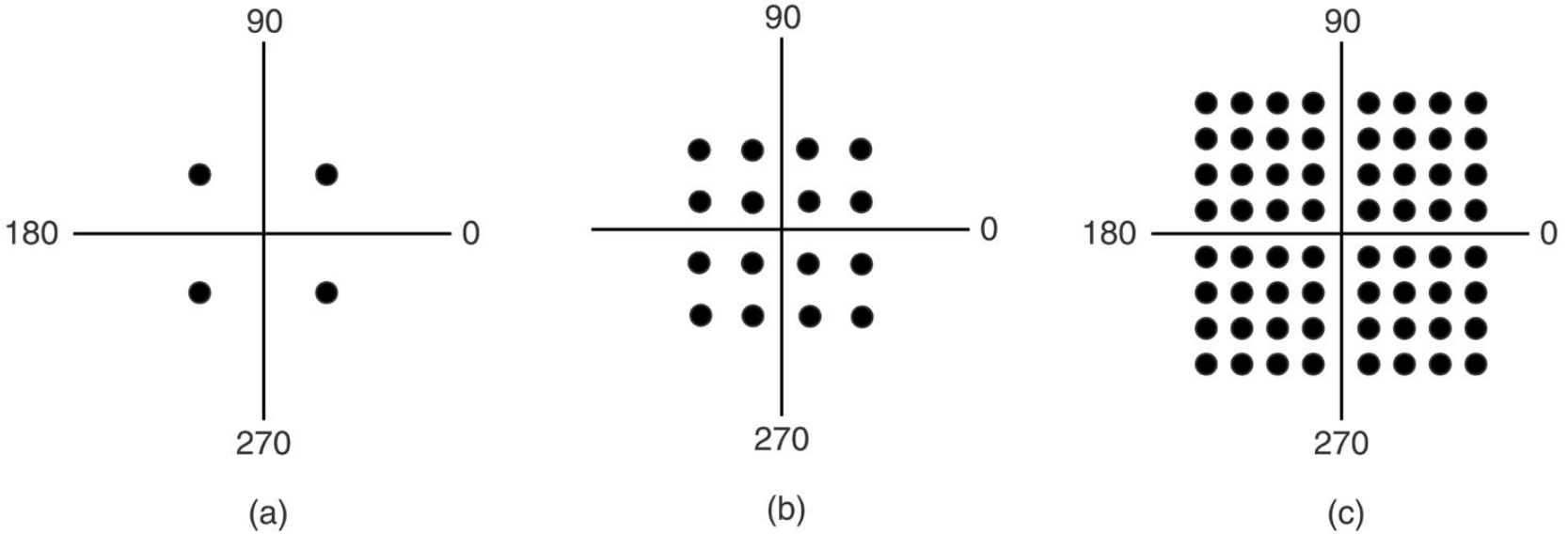
- To visualize a QAM modulation, we use the constellation diagram.
- Each combination of amplitude and phase shift is viewed as vector which is plotted on an x-y grid.
- If we plotted all the possible combinations of amplitude and phase, we'd have a lot of vectors pointing out from the center. So many, in fact, that it would be hard to see all of them.



Quadrature Amplitude Modulation (QAM)

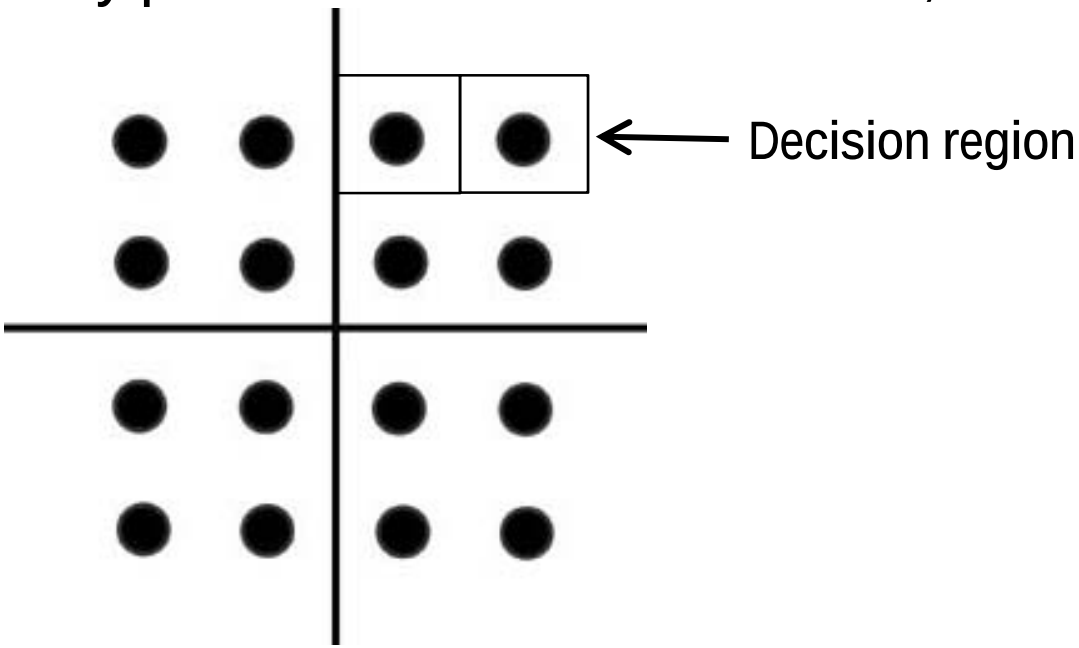


- So, what we do is just indicate, with a dot, the end point of each vector .
- The result is called a signal constellation.
- Below, we have a 4-point, a 16-point, and a 64-point constellation.
- The ITU V.22bis modem was the first “standard” modem to use QAM (2400bps, 600 baud, 16-point constellation).



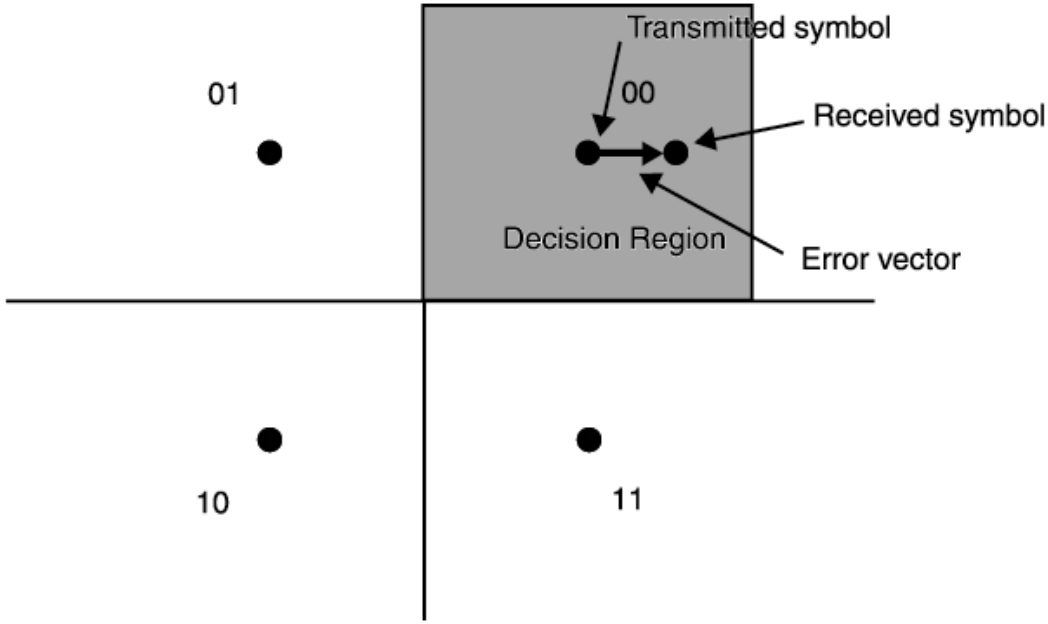
Quadrature Amplitude Modulation (QAM)

- Noise in the channel can affect the amplitude and/or the phase of a constellation point, causing it to “move”.
- The receiver has a “decision region” for each point, and as long as the point is within the decision region, the point will be interpreted correctly.
- This limits how many points can be in a constellation, and thus the bit rate.



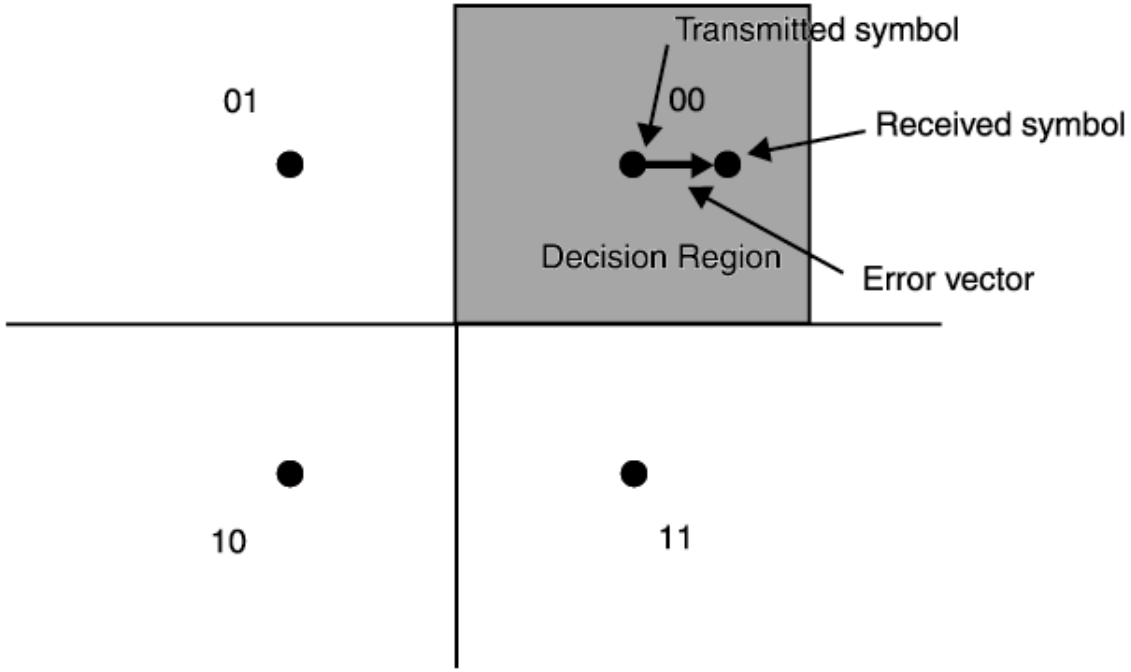
Quadrature Amplitude Modulation (QAM)

- The difference between where the constellation point should be and where it's received is called the "error vector".
- The error vector has been used to implement a secondary channel for monitoring leased line modems.
- A "robust" point is chosen and is purposely sent off position to create an error vector.



Quadrature Amplitude Modulation (QAM)

- An error vector in one direction may mean a 1 while an error vector in the other direction may mean a 0.
- The bit may be sent three times and voting (2 out of 3) is used to choose the bit.
- Very slow channel, perhaps 100bps.

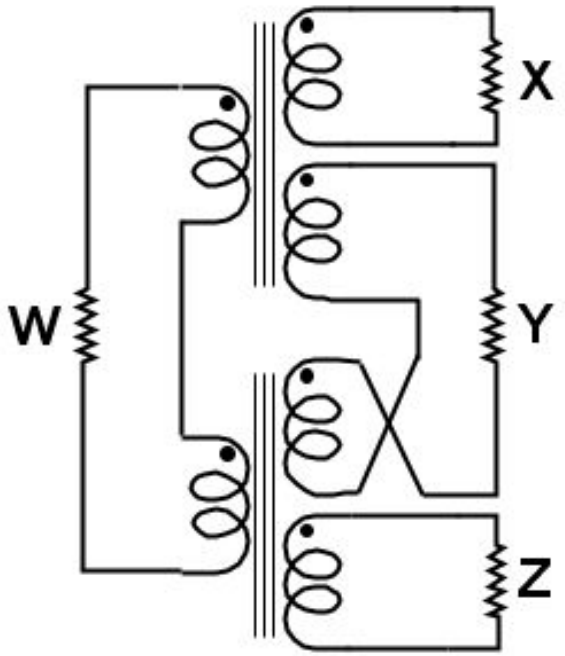


Echo Cancellation

- Prior to the V.32 recommendation, all full duplex dial modems used frequency separation for transmit and receive, which limited the data rate.
  - Bandwidth available for transmit was less than half the channel bandwidth. Usually allowed only 600 baud modems.
- V.34, adopted in 1988 introduced two revolutionary concepts, echo cancellation and trellis coding.
- We'll look at each in turn.

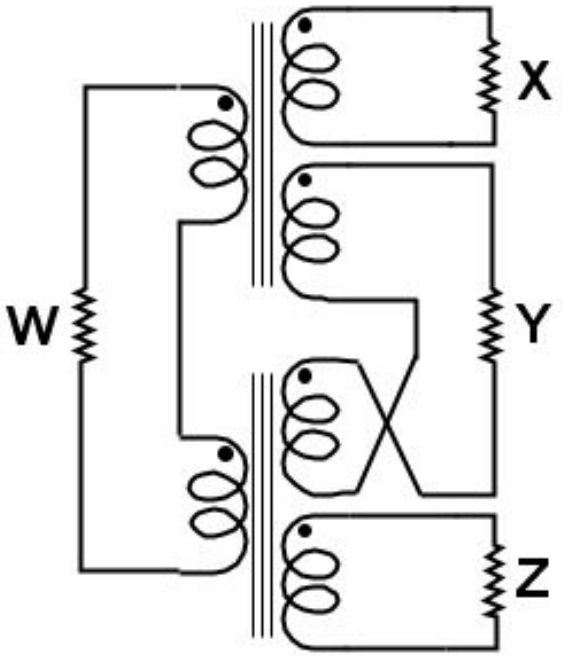
Echo Cancellation

- Earlier I talked about how a hybrid is used between the 4-wire portion of the network and the 2-wire local loop.
- The hybrid has the characteristic that a signal applied at X will appear at W and Y but not at Z.
- A signal applied at W will appear at Y and Z but not at X.
- Requires impedance matching between the line and the hybrid so usually some signal “leaks” to the other port.



Echo Cancellation

- Assume that X is the modem transmit and Y is the modem receive.
- Some of the modem transmit signal will “leak” to the modem receive channel.
- On training, the modem will determine the amount of signal leaking into its receive side.
- From then on, the modem will subtract the amount of leakage from the receive signal, leaving only the signal received from the network.
- Requires a fast DSP processor.



Echo Cancellation

- Opened up the entire voice channel bandwidth for both transmit and receive.
- V.32 used 2400baud (and 2400Hz) with 5 bits/symbol to give 9600bps, full duplex. The carrier was centered at 1650Hz.
  - 4 bits/symbol were data bits and 1 bit/symbol was used for trellis coding. Used a 32 point constellation.
- V.32bis extended V.32 to 7 bits/symbol to give a maximum rate of 14,400bps. Used a 128 point constellation.

Trellis Coding

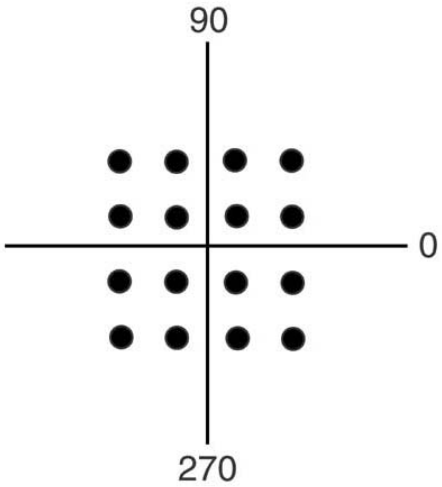
- Trellis coding is a method of correcting some bit errors in the data stream.
- As implemented in modems, it provides the equivalent of a 3dB gain in SNR.
- It is complex and difficult to explain. I'll give a "logical" explanation that is not mathematically correct but will give you the basic concept.
- To explain trellis coding I'd need to explain convolutional coding, set partitioning, and Viterbi decoding. If you're interested, I can give you some links which explain it.

Trellis Coding

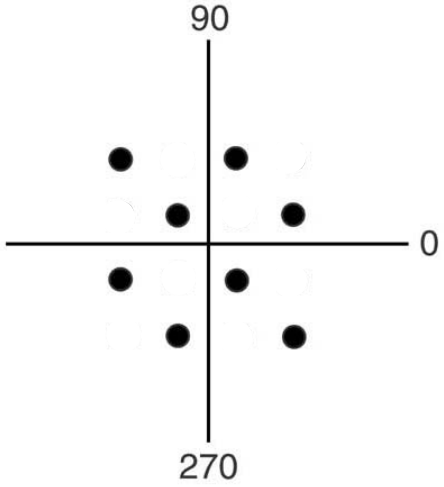
- As we increase the number of constellation points it becomes difficult for the receiver to accurately discriminate which point was actually sent – primarily because of noise in the channel.
- Suppose, however, we had some magic way of removing every other constellation point for the receiver when it makes a decision on what point was actually sent.
- We can do that by partitioning the constellation into two constellations and indicating which is to be used by sending an additional bit.

Trellis Coding

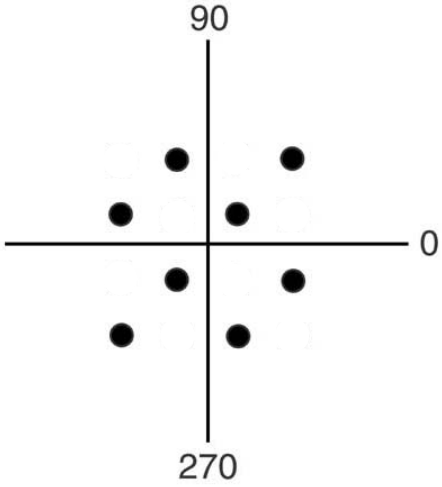
- Here's a visual example of breaking a constellation into two subsets.
- Note that the space between the points is greater in each of the subset constellations than in the original constellation.
- The receiver will have a greater probability of decoding each point accurately, which is equivalent to a better SNR.



Original constellation



First subset constellation



Second subset constellation

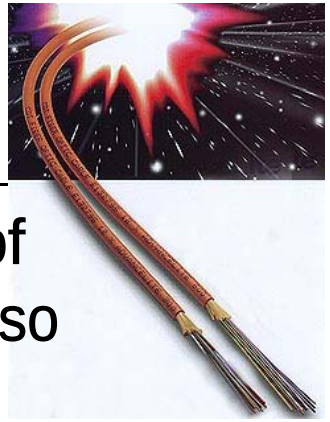
Trellis Coding

- Remember that a V.32 modem has a symbol rate of 2400 symbols/sec and operated at 9600bps.
- This would normally mean that each symbol carried 4 bits (2400 times 4 = 9600).
- But we know that each symbol carried 5 bits – 4 data bits and one bit for trellis coding.
- We can use that one extra bit to decide which constellation subset to use. This will make the decision region larger, making it easier for the receiver to correctly decode the data.
- This is not really how trellis coding works but a detailed explanation would require a fair amount of mathematics.

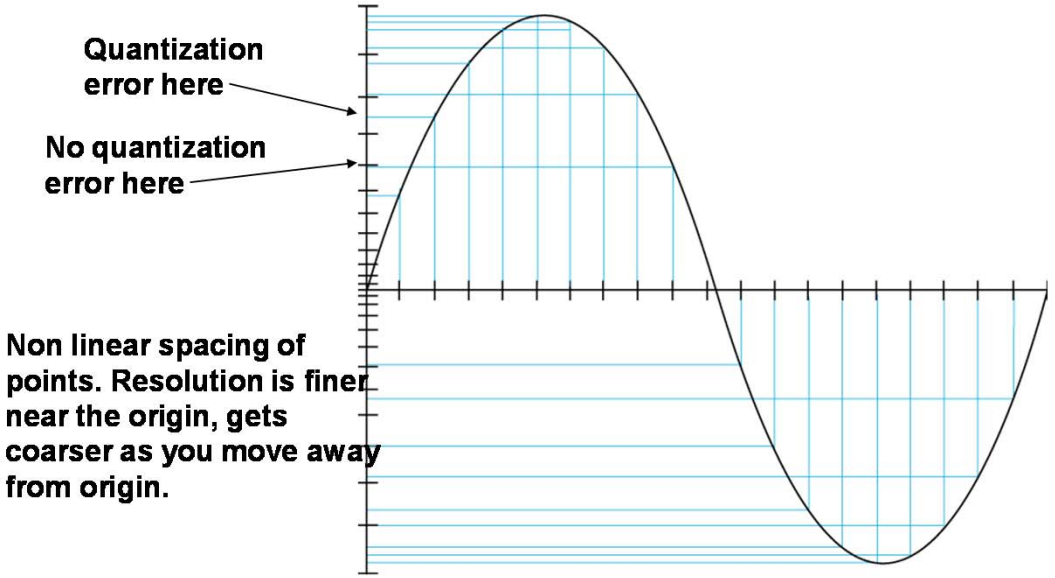
56Kbps Modems

- The telephone system uses 64Kbps to carry a call (or a modem connection).
- The subscriber loop (twisted pair) can carry over 1Mbps.
- So why can we only get 33.6Kbps from a modem (V.34)?
- The answer has to do with the way we convert the signal to digital, using Pulse Code Modulation (PCM), which introduces quantization noise, and with a theorem published by Claude Shannon in 1948.

56Kbps Modems



- Remember when we talked earlier about PCM coding of voice I described the problem of quantization noise (also called quantization error), measured as Signal to Quantization Noise Ratio (SQR).
- For a maximum sinusoidal signal, the theoretical SQR of the μ-law codec is 39.3dB.



56Kbps Modems

- Shannon developed an equation which gives the maximum bit rate through a channel with a certain SNR. The equation is

$$bps = BW\ log_2(1 + \frac{P}{N})$$

Where:
BW= channel bandwidth
P= signal power
N= noise power

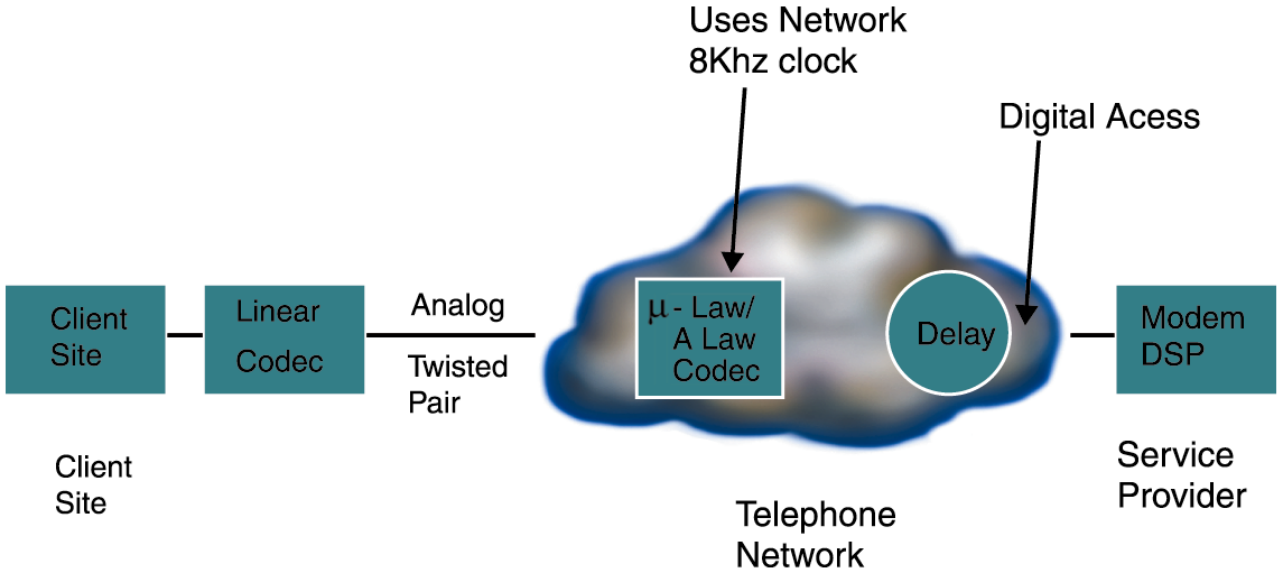
- Note that the signal power and noise power are actual units (Watts) and not dB. Converting the equation to dB, we get

$$bps = BW\ log_2(1 + 10^{\frac{dB}{10}})$$

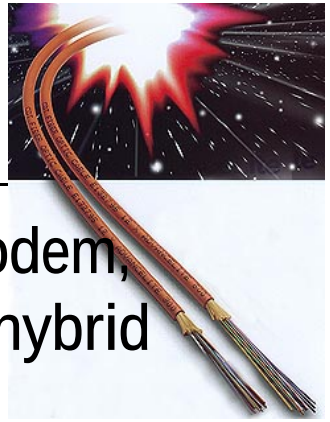
- Using 3000Hz bandwidth and substituting, the maximum bps for a modem is about 39Kbps.
- So how do we get 56Kbps? By eliminating the quantization noise. Let's look at how we do that.

56Kbps Modems

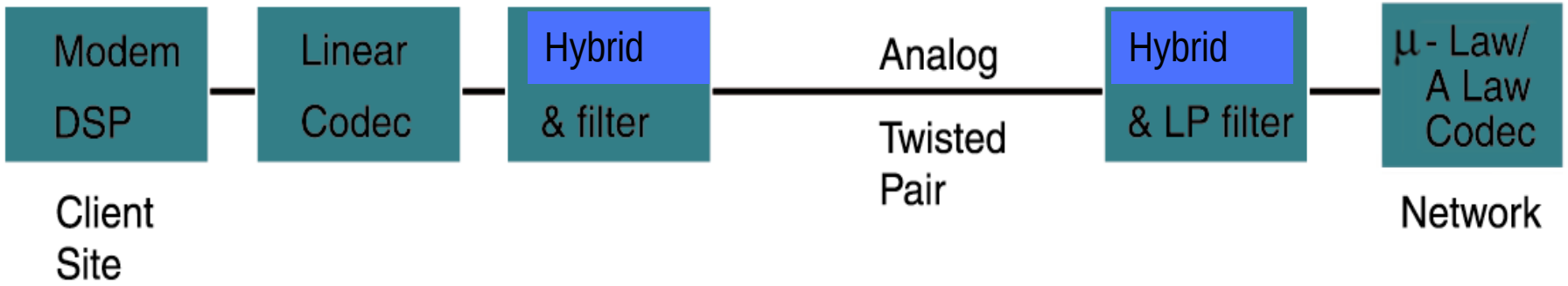
- The 56Kbps modem takes advantage of the fact that the network is all digital.
- To work, the service provider modem has to have a digital connection to the network, such as a T1 line. That way, the service provider's modem can output digital codes which are carried to the codec serving the customer's local loop.



56Kbps Modems



- Looking closer at the connection to the customer's modem, we see that there's a PCM codec in the network and a hybrid to convert from 4-wire to 2-wire on the network side.
- At the customer's modem, we have a hybrid and a linear codec which converts the analog signals to digital to be processed by the Digital Signal Processor (DSP).
- The codec in the network can output 255 different voltage levels. So the signal from the network to the customer's modem is Pulse Amplitude Modulation (PAM), a stream of pulses with different amplitudes.



56Kbps Modems

- So how many different signaling levels do we need for 56Kbps?

$$N_s = 2^{\frac{bps}{sig}}$$

Where:
N_s= Number of symbols
Bps= bits per second
Sig= signaling rate

- The signaling rate is 8,000 times per second, so we need 56,000/8000 or 7.

$$N_s = 2^7$$

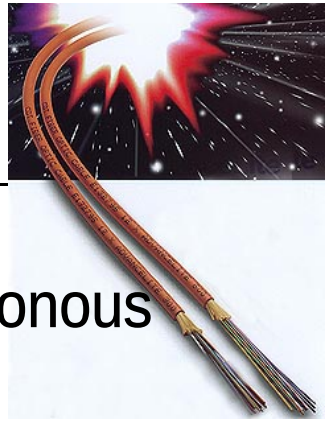
$$N_s = 128$$

- So only 128 of the 255 levels of the codec are needed to produce 56Kbps. This modem technique can be described as a 128PAM technique.

56Kbps Modems

- If the modem were to drop back to 48Kbps, only 64 levels would be needed (2^6).
- Trellis coding is not used in the downstream direction in the 56Kbps modem. Can you guess why?
- Other communication services use PAM, for example ISDN BRI uses a technique called 2B1Q which is a four level PAM technique, so PAM communication is well understood.
- However, there are some important problems in communicating 56Kbps in this manner, which I've ignored in this discussion.

V.42 Error Correction



- Almost all dial modems were used to connect asynchronous terminals to a computer.
- With async, we can't tell if an error occurred in the transmission. So, you might have typed a "B", which is 0100 0010, but the computer might have received a "C", which is 0100 0011.
- One way to provide reliable communication is to put a number of characters into a packet and send the packet with a code that detects if errors occurred.
- If an error occurred, the packet can be re-transmitted.
- We're now getting into the area of protocols.

Protocol Frames

- There are many different protocol frames used in communications but, for all of them, there must be a way to identify the start of the frame, and a way to tell when the frame ends.
  - Start and end flags, such as are used in HDLC.
  - Fixed length frames which only have a start indicator, such as ATM cells.
  - Variable length frames with a start flag and a length indicator.
- For synchronous communications, there must be an idle character which is continuously transmitted when there's no data to be sent. This is required to keep the receive clock synchronized.

HDLC Frame

- An HDLC frame consists of the following fields:
  - Start flag. These are the bits 0111 1110 or 0x7E.
  - Address field
  - Control field
  - Data
  - Frame check. A 16 bit or 32 bit Cyclical Redundancy Check (CRC). This is used to detect if errors occurred in transmission of the frame. Wikipedia has a good discussion of CRC.
  - End flag or final flag. The bits 0111 1110, same as the start flag.
- Continuous flags are sent when the line is idle.

Flag (7E)	Address (8bits x n)	Control (8bits x n)	Data	FCS (16/32 bits)	Flag (7E)

V.42 Error Correction

- A problem with HDLC is that a special character is used to frame the packet – 0111 1110 – which has six 1 bits in a row, so something has to be done to prevent the appearance of that character in the data.
- A technique known as “Zero Bit Insertion” is used.
- After five consecutive 1 bits a 0 bit is inserted.
  - This is true even if the next bit is a zero.
  - Typically occurs about every 32 payload bits.
- This causes an increase in the number of bits in the message.

V.42 Error Correction

- V.42 works by taking the async characters from the terminal interface (DTE), putting them into the data field of an HDLC frame, computing a CRC, and sending the frame to the other modem.
- The receiving modem does its own CRC calculation, compares it to the transmitted CRC, and if no errors occurred, feeds the characters, in async form, to the DTE interface.
- If the CRCs don’t check, the receiving modem sends a request to re-send the frame.
- Packetizing the data introduces a delay in the transmission path.
- When there’s no data to be sent, continuous flags are sent.

V.42bis

- Much of the data communicated via modem is not completely random and is therefore compressible.
- Various algorithms have been developed which can take non-random data and encode it in less bits than the original source.
- One algorithm is the Lempel-Ziv algorithm, a variant of which is used in V.42bis.
- We have the perfect opportunity to compress the data sent via a modem which implements V.42 because V.42 will buffer the data into blocks.
- This can provide very high communication rates for compressible data.

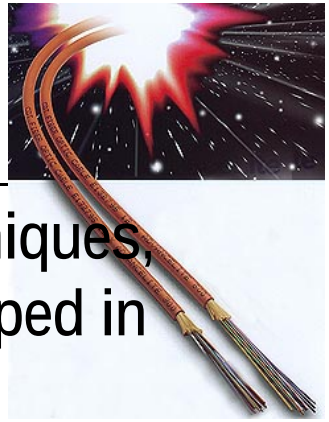
Problems with Dial Modems

- Voice line modems are limited by the bandwidth of a voice channel, thereby limiting their data rate.
  - The 56Kbps modem is a fast is as voice line modem is going to get.
- Long connection establishment time (dialing, call setup and training).
- Cannot even know if system has data for you until you connect.
- Dial up modems block the use of the line for making or receiving voice calls while they are in use.

Digital Subscriber Line (DSL)

- Telephone line modems are limited by the relatively small amount of bandwidth available – 3000Hz to maybe 3400Hz.
- If we could use more bandwidth, we could operate at a much higher bit rate.
- To do that, we cannot go through the traditional telephone system – we must bypass that network.
- How can we do that?
- One way is to terminate the subscriber loop at the network end with a high speed modem and to put an equivalent modem at the customer location.

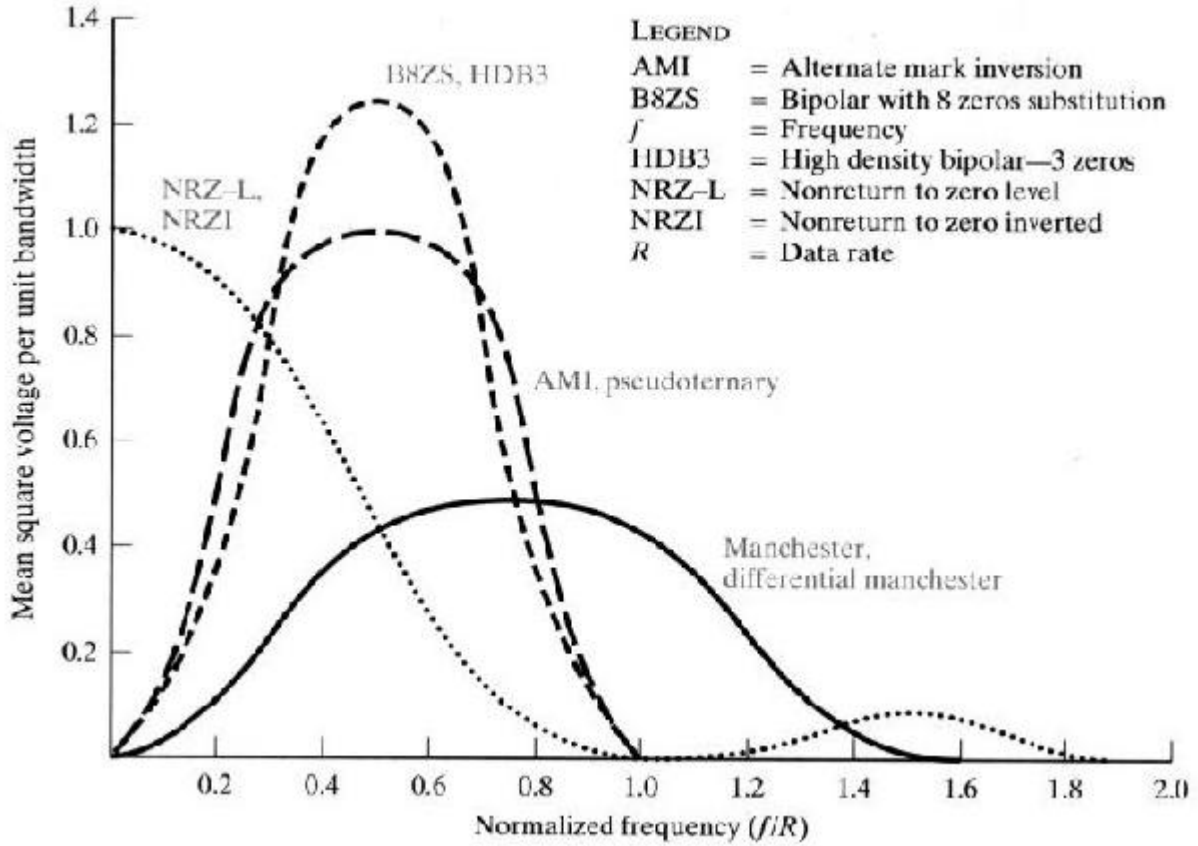
High Speed Digital Subscriber Line (HDSL)



- Before I look at some of the more advanced DSL techniques, let me look at the first DSL technique – HDSL – developed in 1993.
- T1, using AMI, was provisioned over two twisted pair – one for transmit and one for receive.
- T1 lines, using AMI, had a power spectral density with its first null at 1.544MHz, and significant side lobes. Also, lots of NEXT to other lines.
- The twisted pair used in the telephone network was ill suited to carry this signal - too much high frequency attenuation.
- This required regeneration of the T1 signal every 6,000 feet (a bit more than a mile).
- Provisioning of a T1 line was very slow because these repeaters had to be installed along the path of the circuit.

HDSL

- Here's a power spectral density plot for various line codings.
- Note that AMI has it's first null at 1.544MHz (frequency over rate is 1).

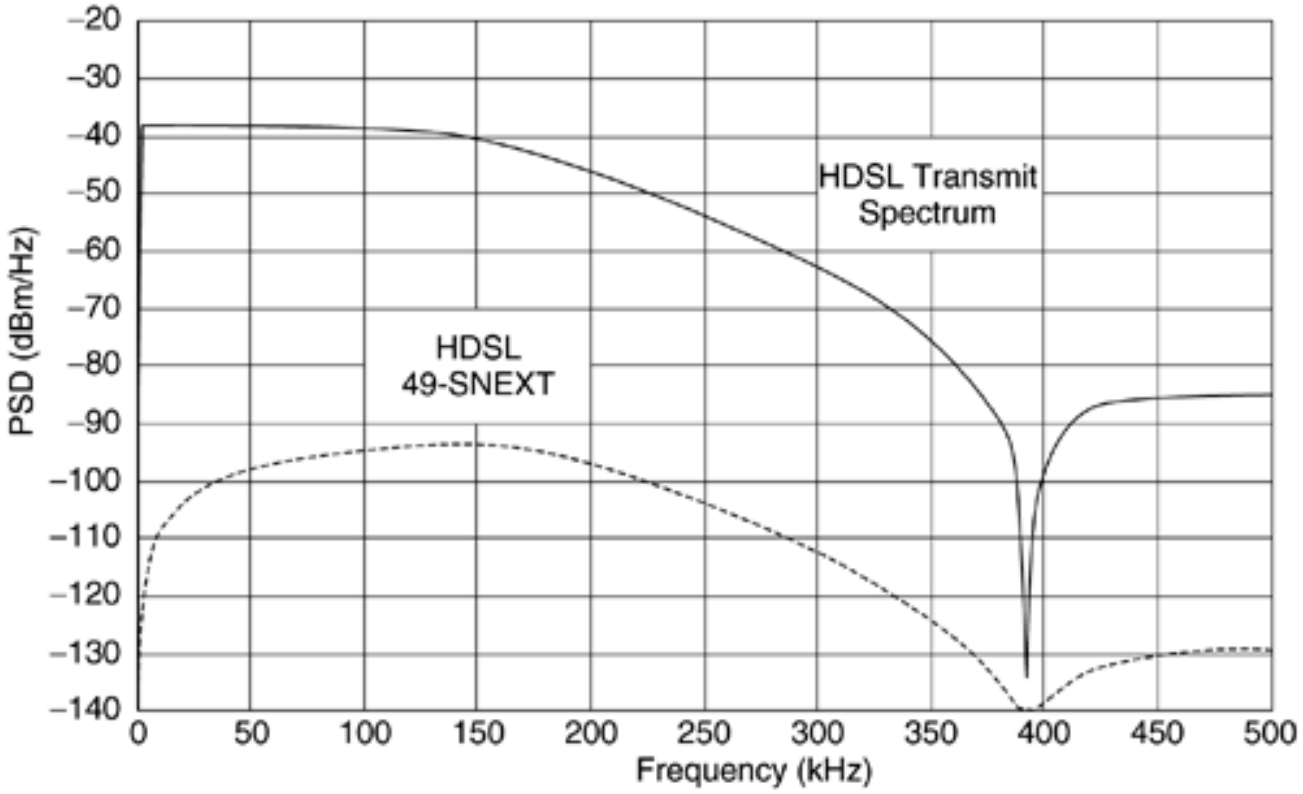


HDSL

- The first HDSL took the 2B1Q line code and used it to provide a longer reach for provisioning a T1 line.
- The 1.544Mbps line was divided into two full duplex circuits (using echo cancellation) – so four of the HDSL modems were required to provision one T1 line.
- Each line carried 784Kbps:
  - The 1.536 payload was divided in half (768Kbps).
  - Each line carried the framing bits (8Kbps)
  - Framing was added to each line to demux the data (8Kbps).
- At the receive end, the data was merged back together to form the original T1 line.
- Allowed provisioning to about 12,000 feet and could use repeaters for longer reach.

HDSL Power Spectral density

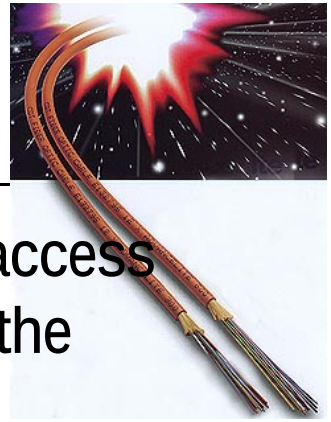
- Note that the first null for HDSL occurs at half the line rate, or at 392Kz.



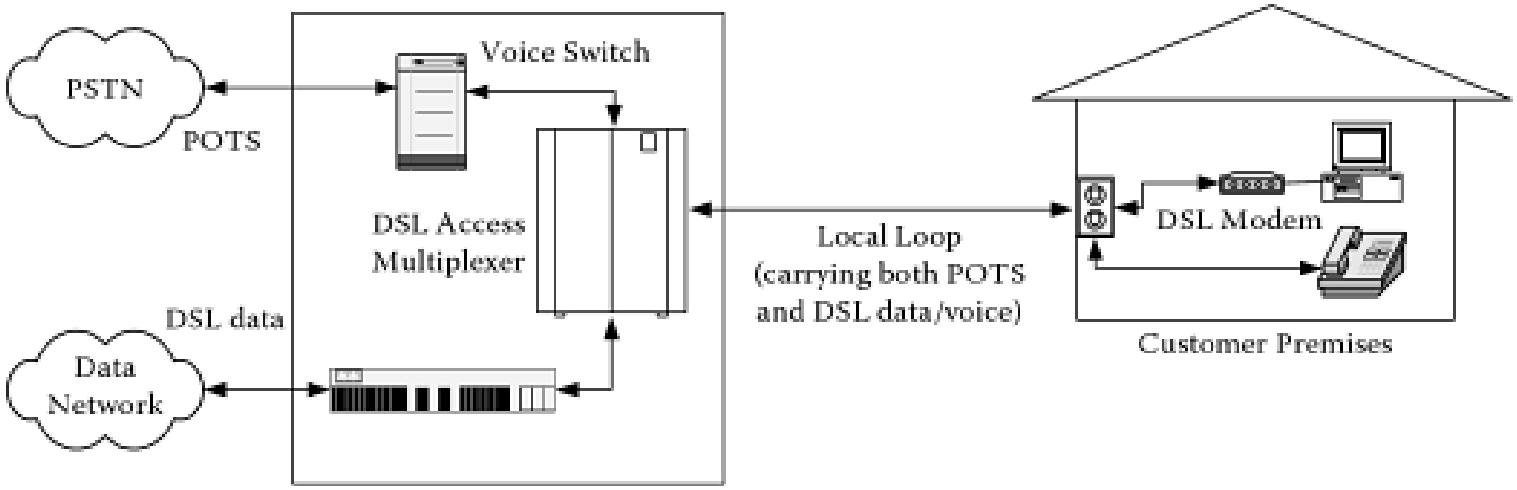
Other Flavors of HDSL

- HDSL – The ITU standardized HDSL in recommendation G.991.1
- HDSL2 – (G.991.2) 1.544Mbps over a single pair, full duplex. Uses 16 level PAM with trellis coding – 3 information bits and 1 trellis coding bit per baud. Symbol rate is 517.33Kbaud. Reach is about 9,000 feet
- HDSL4 – 1.544Mbps over two pair, full duplex. Uses 16 level PAM with trellis coding. Each pair carries half the payload, plus overhead for inverse multiplexing. Uses about half the bandwidth of HDSL2. Reach is about 11,000 feet.

Asymmetric Digital Subscriber Line (ADSL)

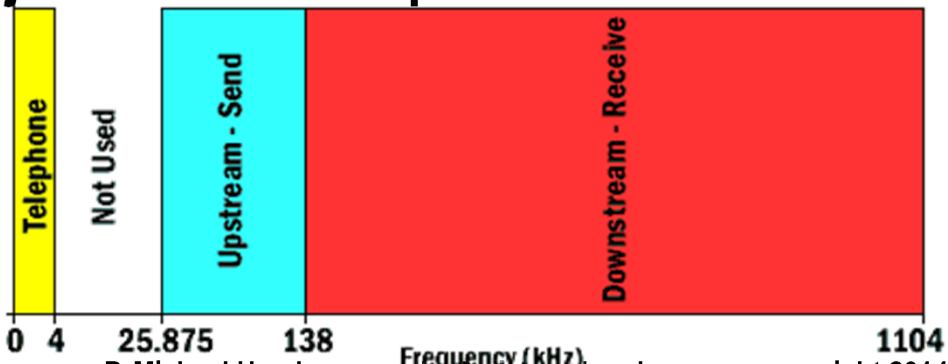


- We can use the local loop to provide high speed data access to the home by using two high speed modems, one at the central office and one at the subscriber location.
- Since the local loop can provide much more than 4KHz of bandwidth, we can operate the modems at much higher rates, perhaps several megabits/sec.
- The modem at the central office would then be connected to a router into the Internet.



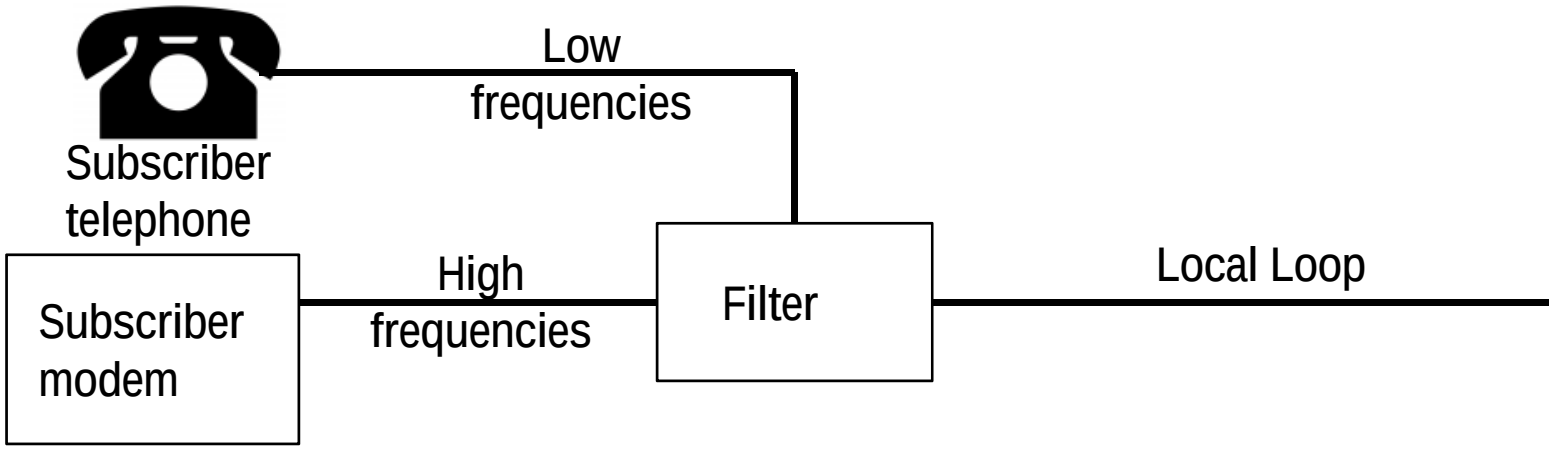
Digital Subscriber Line (DSL)

- But the telephone company didn't have enough extra pair to provide a separate service to their subscribers.
- To solve that problem, the high speed modems are operated in a frequency range above the 4KHz voice channel, so it can be provisioned on the same line as POTS.
- The upstream and downstream are frequency separated.
- The upstream (to the network) band is above the voice band, extending from about 25KHz to 138KHz.
- Downstream is allocated the bulk of the bandwidth and is located **just above the upstream band.**



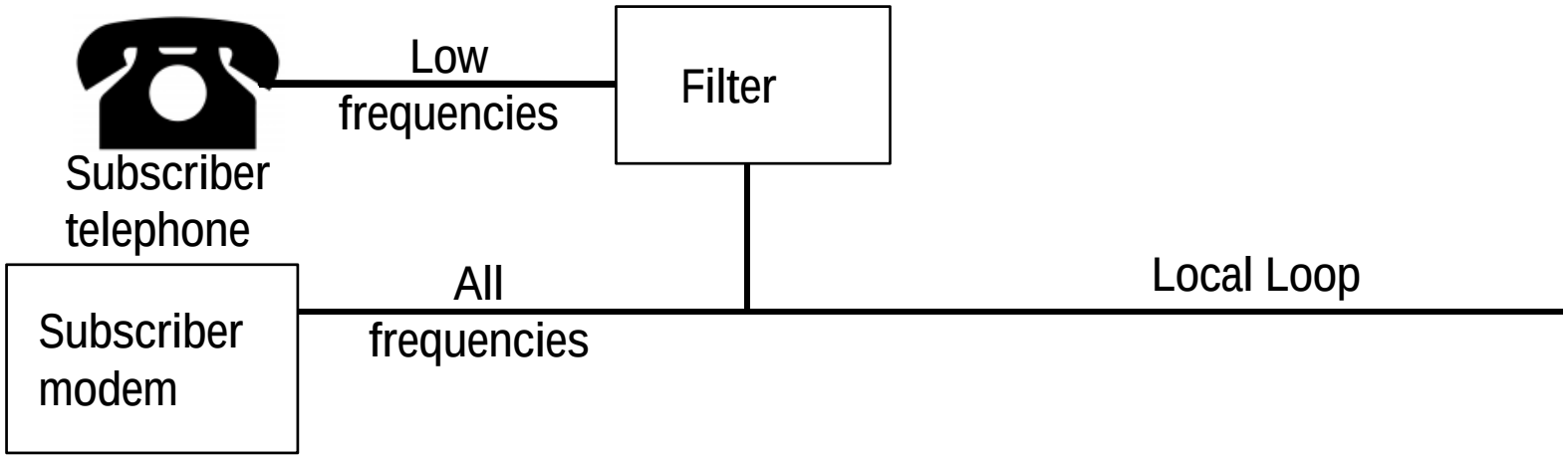
Digital Subscriber Line (DSL)

- To separate the voice from the modem signal, a filter is used on both ends of the link.
- Installing this at a customer's home was expensive. A craft person had to go to the house, install the filter where the local loop entered the home, and run a separate twisted pair to the modem location.



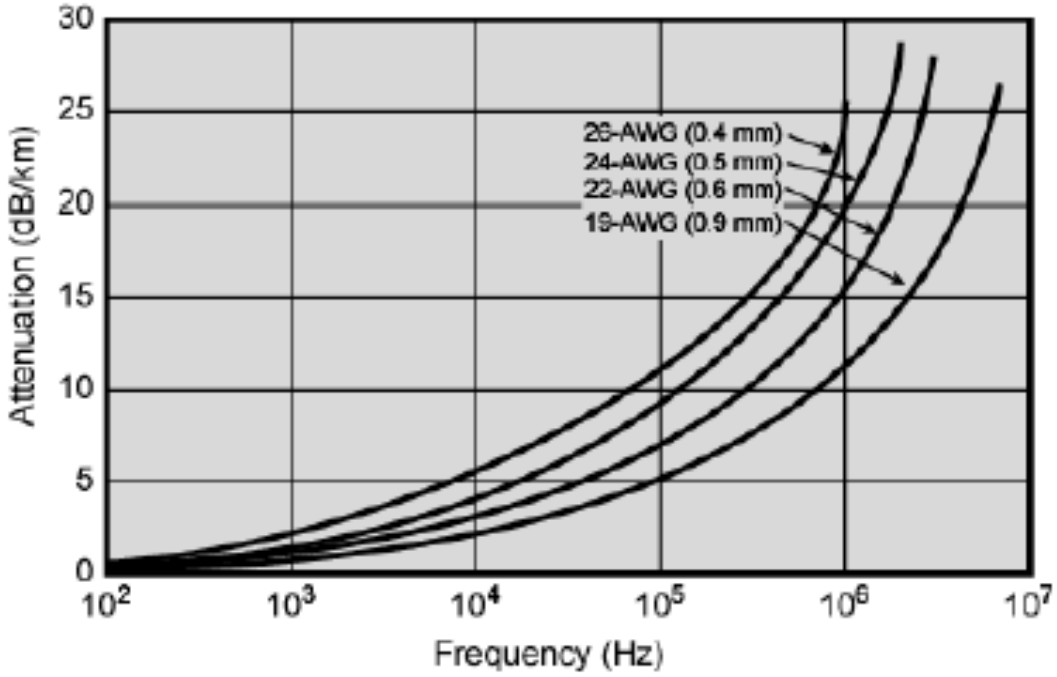
Digital Subscriber Line (DSL)

- Eventually, the phone companies recognized that individual low pass filters could be put on each phone.
- This allowed DSL modems to be customer installed.



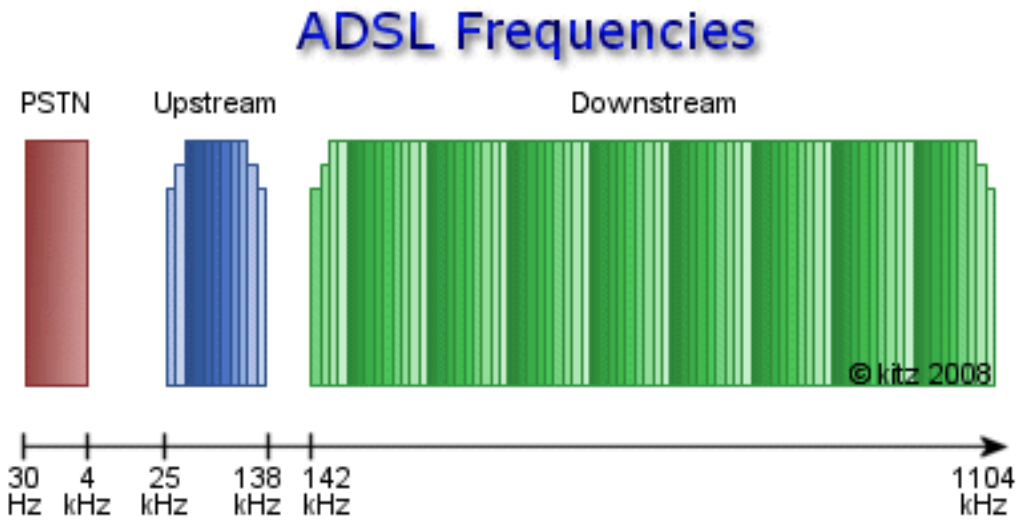
Digital Subscriber Line (DSL)

- The local loop was never designed to carry high frequency modem signals.
  - Only about 7 to 9 twists per meter.
  - Attenuation at higher frequencies (around 1MHz) is very high.
  - Longer loops have more attenuation.
  - NEXT at the central office side makes receiving from the subscriber modem challenging (that's one of the reasons upstream is at a lower frequency than downstream).

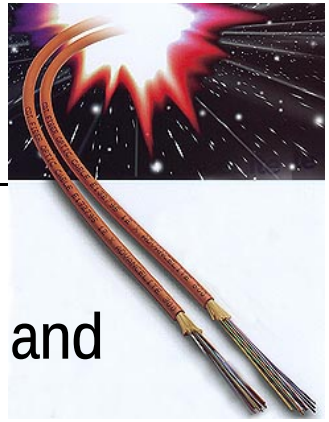


Discrete Multi-Tone

- A different modulation was chosen for ADSL to help deal with the increasing attenuation at higher frequencies – Discrete MultiTone (DMT).
- DMT put many individual channels within the bandwidth. Each channel was 4.3125KHz.
- QAM was used to modulate each carrier, with the number of bits per baud dependent on the quality of that channel.



Discrete Multi-Tone



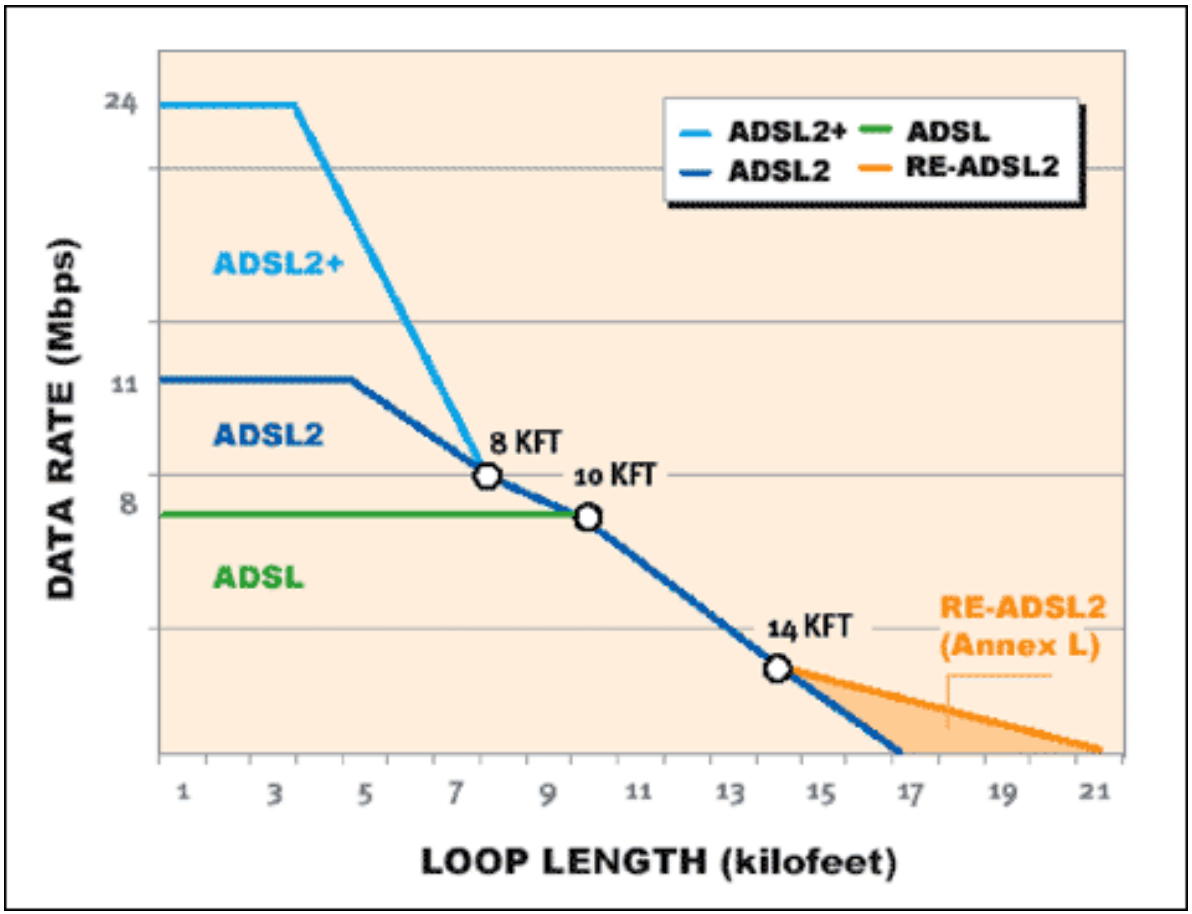
- There are 25 channels (called “bins”) for the upstream and 224 for the downstream. 256 total, starting at 0Hz.
- The number of bits per baud per bin can vary from 2 to 15, depending on the SNR for that bin.
- Because of the higher attenuation at higher frequencies, the high numbered bins will carry few bits/baud – if any at all.
- Two bins are used for pilot tones, bin 16 (69KHz) for upstream and bin 64 (276KHz) for downstream.
- DMT is also known as Coded Orthogonal Frequency Division Multiplexing (COFDM).

ADSL

- The bit rate performance of ADSL is limited by the length of the local loop.
- The first ADSL, (G.992.1) called ADSL1 today, provided a maximum downstream rate of 8Mbps, and a maximum upstream rate of 896Kbps.
- ADSL2 (G.992.3) offers various combinations of upstream and downstream. Normal maximum downstream rate is about 12Mbps. The maximum upstream rate is 3.5Mbps but is usually configured for less.
- ADSL2+ (G.992.5) takes the downstream rate to as high as 24Mbps. The upstream is the same as ADSL2. It achieves these rates by using more bandwidth, with the upper end about 2.2MHz.

ADSL

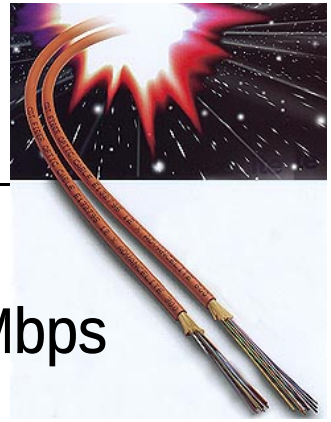
- This chart shows how the three ADSL versions compare for downstream performance by reach.



Very High Rate Digital Subscriber Line (VDSL)

- The telephone companies wished to provide bundled services to their subscribers, specifically television – in competition with the cable companies – in addition to Internet access and telephony.
- HDSL was not fast enough and there was no need for symmetric data rates.
- ADSL was not fast enough to provide a good user television experience.
- VDSL was developed to fill that niche.
- It is significantly limited in reach so it can be classified as a hybrid fiber/copper technology.

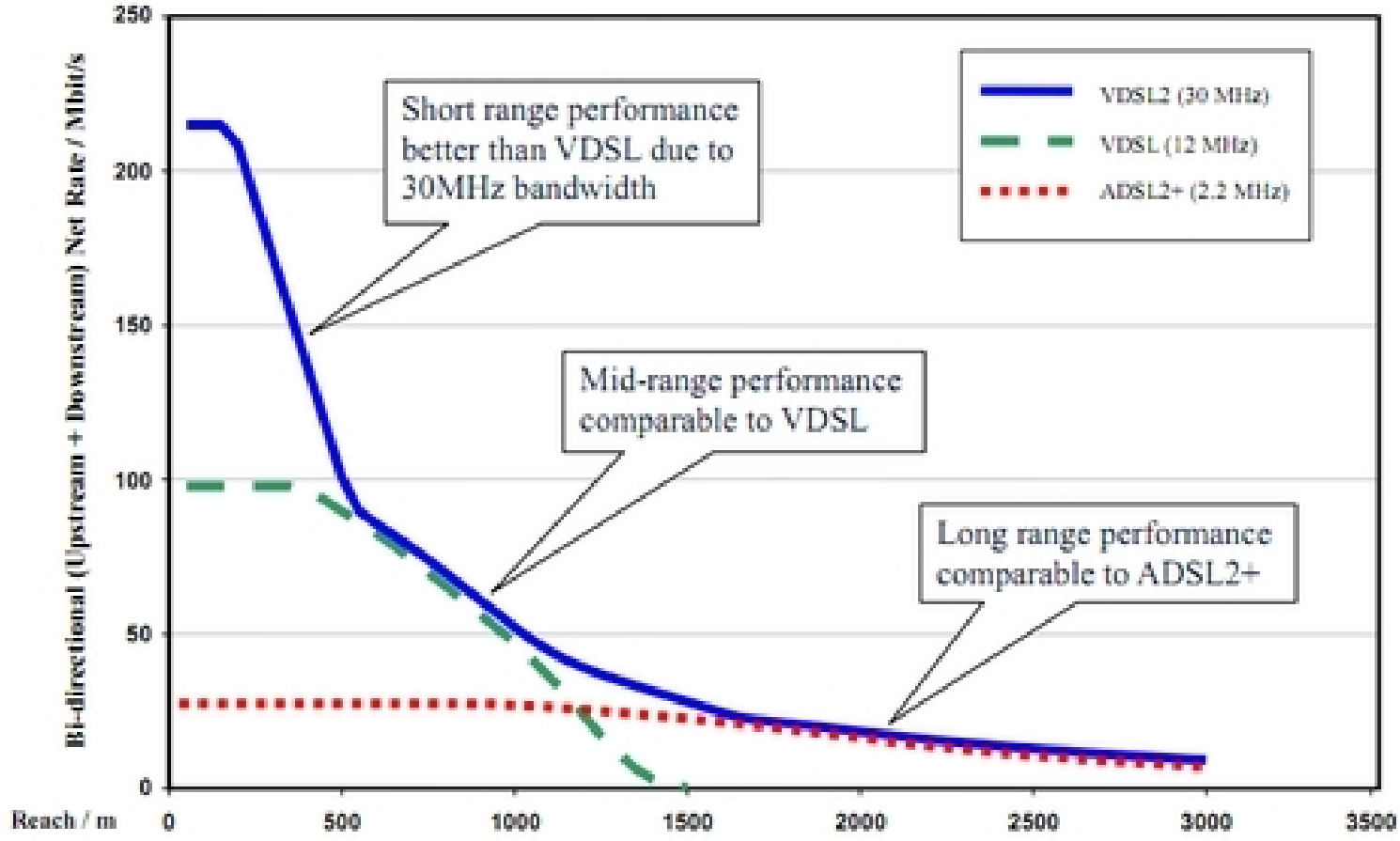
VDSL



- The original VDSL technology (G.993.1) supported 52Mbps downstream and 6.4Mbps upstream, using frequency division on the local loop.
- It used DMT modulation and bandwidth out to 12MHz.
- It can be provisioned over a standard POTS service.
- The follow-up technology – VDSL2 (G.993.2) – can support up to 100Mbps using bandwidth out to 30MHz.
- But VDSL2 can only operate at the highest speed on very short loops, perhaps 500 meters.

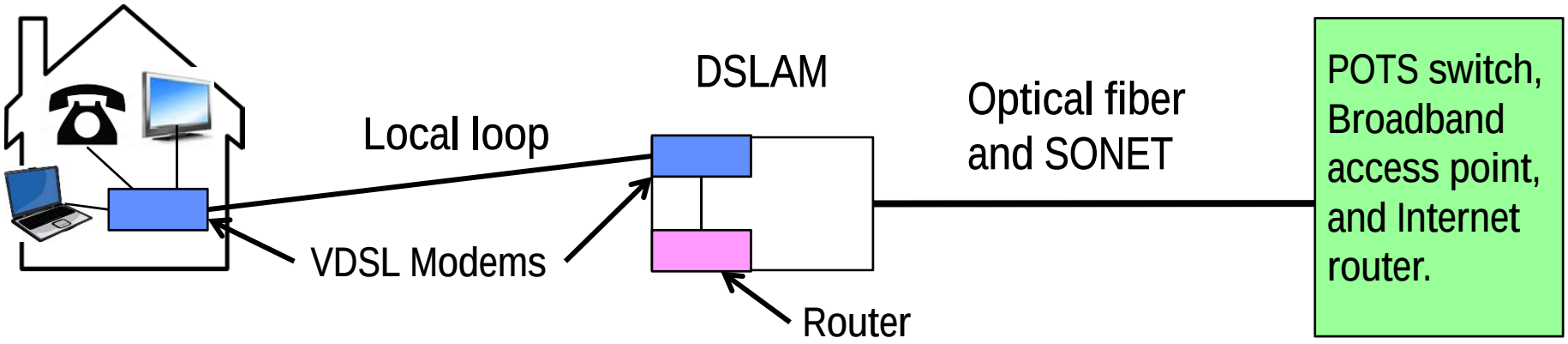
VDSL Rate and Reach

- The chart compares the rates and reach of VDSL, VDSL2, and ADSL2+



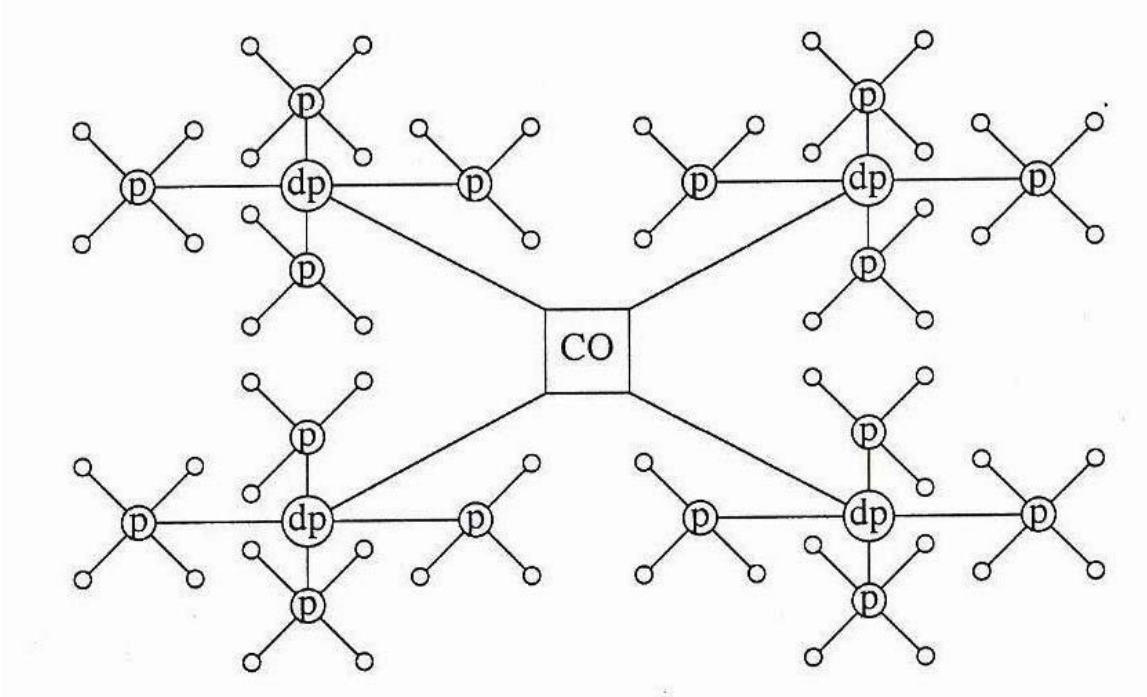
VDSL

- To provide VDSL services, the provider has to “shorten” the local loop.
- This is done by installing a Digital Subscriber Line Access Multiplexer (DSLAM) in the neighborhood.
- The DSLAM provides a line card to terminate POTS, plus a VDSL modem and routing function.
- The connection to the central office, to the Internet, and to the broadband access point is through fiber with SONET as the protocol.



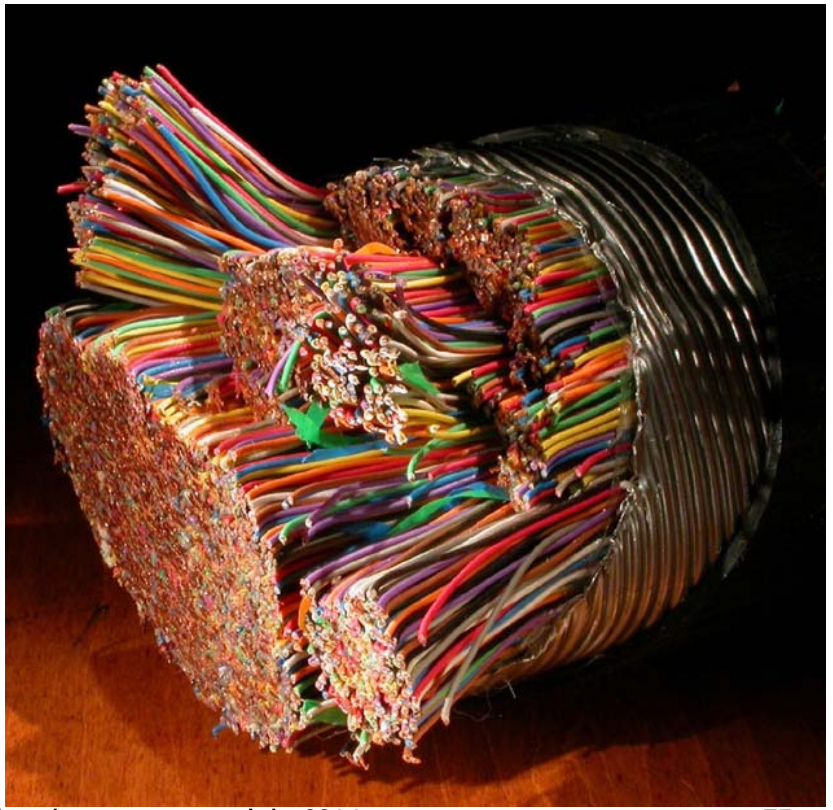
VDSL

- Here's an example of the Telco loop plant topology.
- The TELCO will run fiber from the Central Office (CO) to the Distribution Points (DP) or the Pedestals (P), depending upon how many houses are served from each.
- At the end of the fiber they will install a DSLAM.



Feeder Cable

- And just as an aside, this is what the feeder cable that used to run between the Central Office and the Distribution Points.
- This cable was replaced by optical fiber between the CO and the DSLAM.
- You can imagine what it must have been like for the crafts person who had to connect the proper wires together at the Distribution Point.
  He or she definitely could not be color blind!



DSLAM

- Here's a picture of the DSLAM (on the right) that serves my neighborhood.
- It will almost always be with a pedestal wire box that used to terminate the cable bundle to the neighborhood.

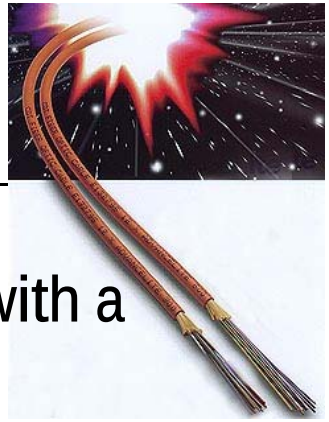
G. fast

- Work is going on in the ITU under the name “G.fast” to enhance VDSL to operate at higher rates, albeit at very short distances.
- G.fast will allow operation with up to 212MHz of bandwidth and across two “bonded” lines.
- Will use DMT and will operate as high as 1Gbps on short loops. 200Mbps to 500Mbps is more likely.
- Targeted primarily at multiple dwelling units (apartment buildings) where fiber can be brought into the basement, then G.fast can be used over existing twisted pair to the dwelling unit but can also be used to homes with “mini-DSLAMs” on the pole.
- Also called Fiber to the Distribution Point (FTTdp).

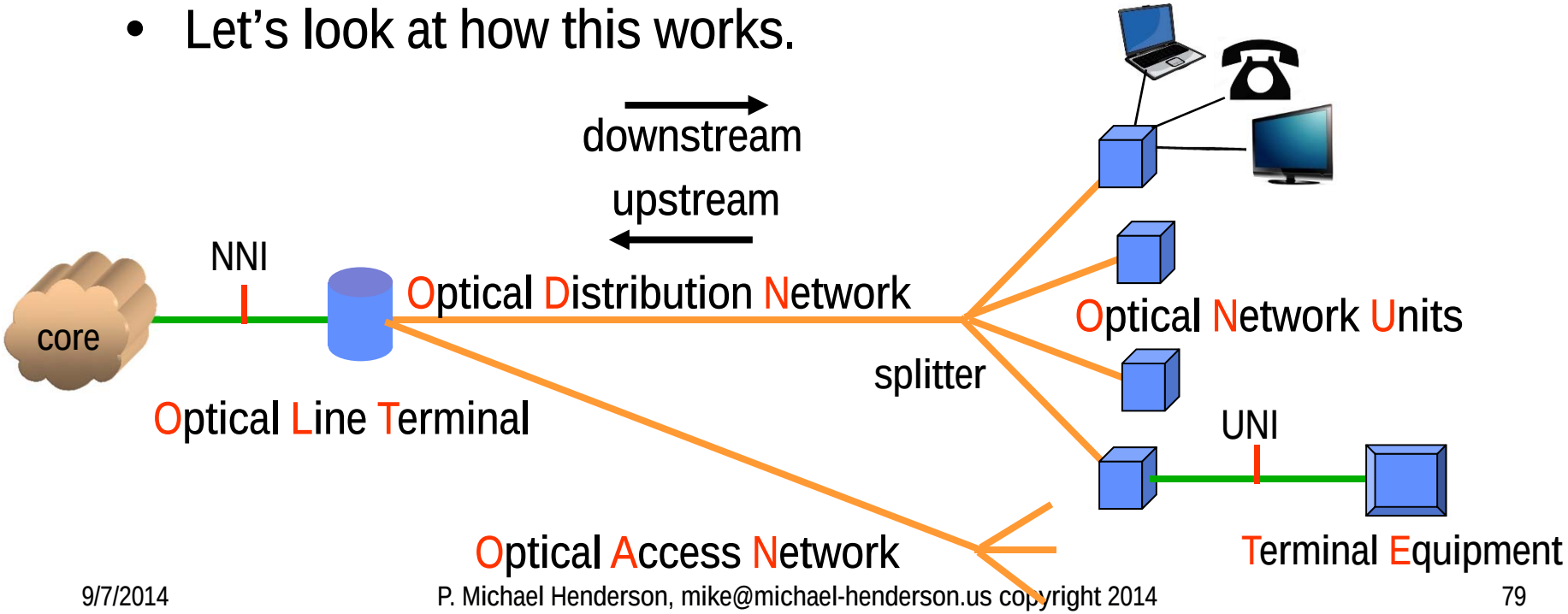
Fiber to the Home (FTTH)

- The local loop twisted pair is a bandwidth limited transmission medium, and the modern DSL modems communicate close to the limits of the wire.
- Optical fiber has significantly greater bandwidth. Why not use "fiber to the home"?
  - While the twisted pair was already in place, optical fiber has to be laid.
  - The components of optical communications are generally of higher cost than for twisted pair.
- But FTTH is being used. Verizon's FiOS is fiber to the home.

Passive Optical Network (PON)

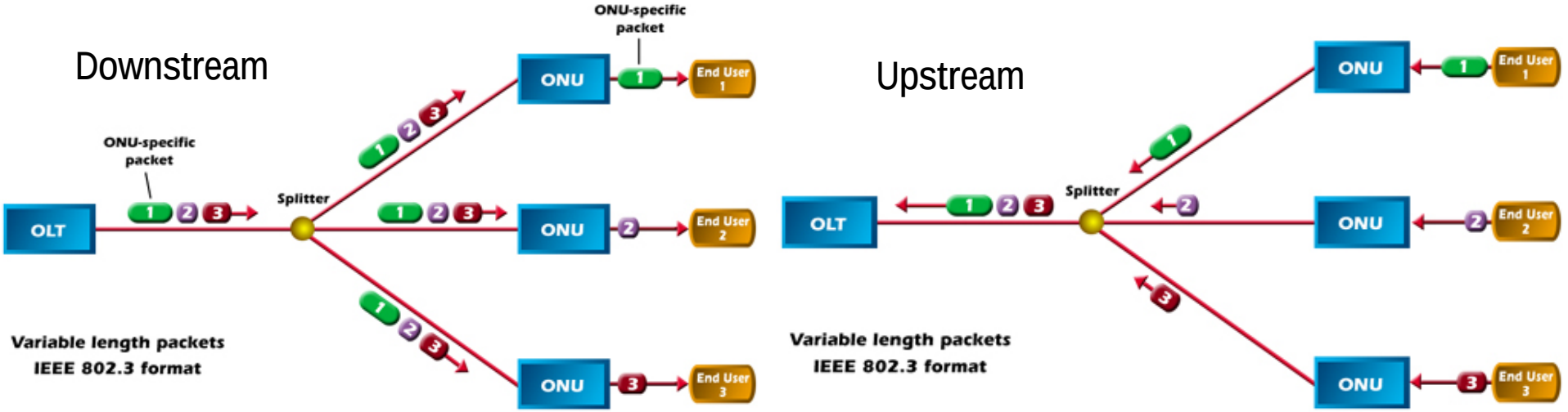


- The general concept of a PON is that the fiber is split with a passive splitter which directs a portion of the light to multiple fibers.
- This is less expensive than running a fiber (or two) from a network node to each subscriber.
- Let's look at how this works.



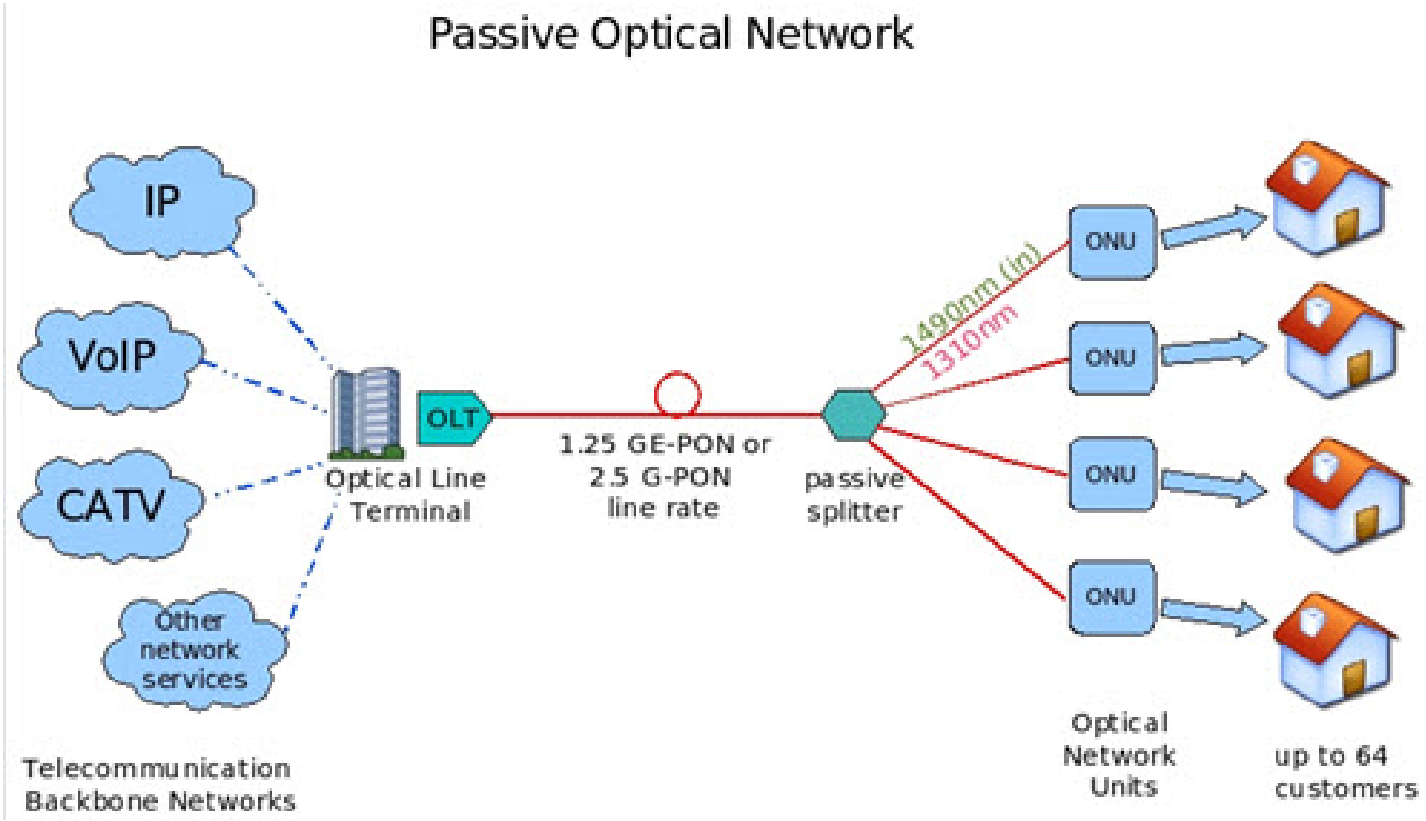
Passive Optical Network (PON)

- Downstream (DS) traffic is sent on 1490nm and upstream (US) traffic is sent on 1310nm.
- DS traffic goes to all ONUs who throw away data that is not directed to them.
- US traffic is TDM. Each ONU is given a time slot to send traffic.
- GPON rates are 2.48832Gbps down and 1.24416Gbps up .



Passive Optical Network

- For residential customers, the standard offering is what's known as "triple play": phone, Internet access, and television.



Flavors of PON

- A lot of specifications for passive optical networks have been developed:
  - APON – ATM PON
  - BPON – Broadband PON - ITU-T G.983.x
  - CPON – CDMA PON
  - EPON – Ethernet PON - IEEE 802.3-2005 clauses 64 and 65
  - GPON – Gigabit PON - ITU-T G.984.x
  - GEPON – Gigabit Ethernet PON
  - WPON – Wavelength division multiplexing PON
- We’re mostly going to look at GPON here.

GPON Specifications

- Maximum length of fiber from OLT to ONU is 20km. A long reach version can operate to 60km.
- Downstream traffic is sent on 1490nm and upstream traffic is sent on 1310nm over single-mode fiber.
- GPON rates are 2.48832Gbps down and 1.24416Gbps up. Usually spoken of a 2.5Gbps down and 1.25Gbps up.
- Downstream is synchronous. Idle characters are sent if no data.
- Maximum fiber split is 1:64
- Can carry voice, data, and video encapsulated in GPON Encapsulation Method (GEM).

Passive Optical Networks

- There are a number of functions in passive optical networks:
  - Auto discovery.
  - Timing for upstream transmission. Each leg from the splitter is of different length and thus has a different propagation time.
  - Security. All data goes to all users. The data must be encrypted in a secure fashion that only the intended receiver can see it.
  - Dynamic bandwidth allocation upstream.

Autodiscovery

- Every so often, the OLT will send out a “Serial Number Request” message.
- Unregistered ONUs will use a random time period to respond with their 64bit serial number.
- OLT sends a 1 octet ONU_ID to the ONU.
- Ranging is performed to determine round trip time and laser power level.
- ONU enters operational state.

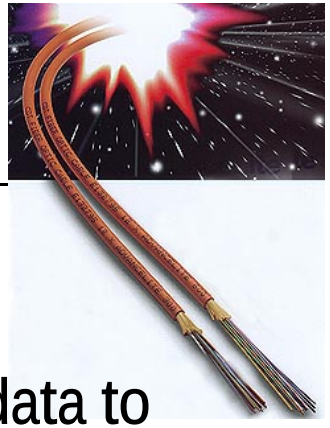
Timing for Upstream Transmission

- In normal operation, the ONUs will receive a “time map” which allocates time for them to transmit upstream.
- But each ONU will be on a different length of fiber so the round trip time (RTT) must be determined.
- During autodiscovery, the OLT and the ONU will perform a handshake which will allow the determination of the RTT.
- The OLT will use the RTT when providing grants for that ONU to transmit upstream.

Security

- During autodiscovery, the OLT and the ONU will negotiate an encryption key for AES-128 encryption.
- The negotiation is secure, similar to https key negotiation in the world wide web.

Dynamic Bandwidth Allocation



- The OLT sends frames to each ONU, even if it has no data to send to it. The frame contains the time slot information for the ONU to send data upstream.
- The ONU maintains different buffers for different kinds of data (which have different priorities) and replies to the OLT with information about the status of the buffers.
- The OLT will use that data to modify the time slot allocations for the ONU.

PON Challenges

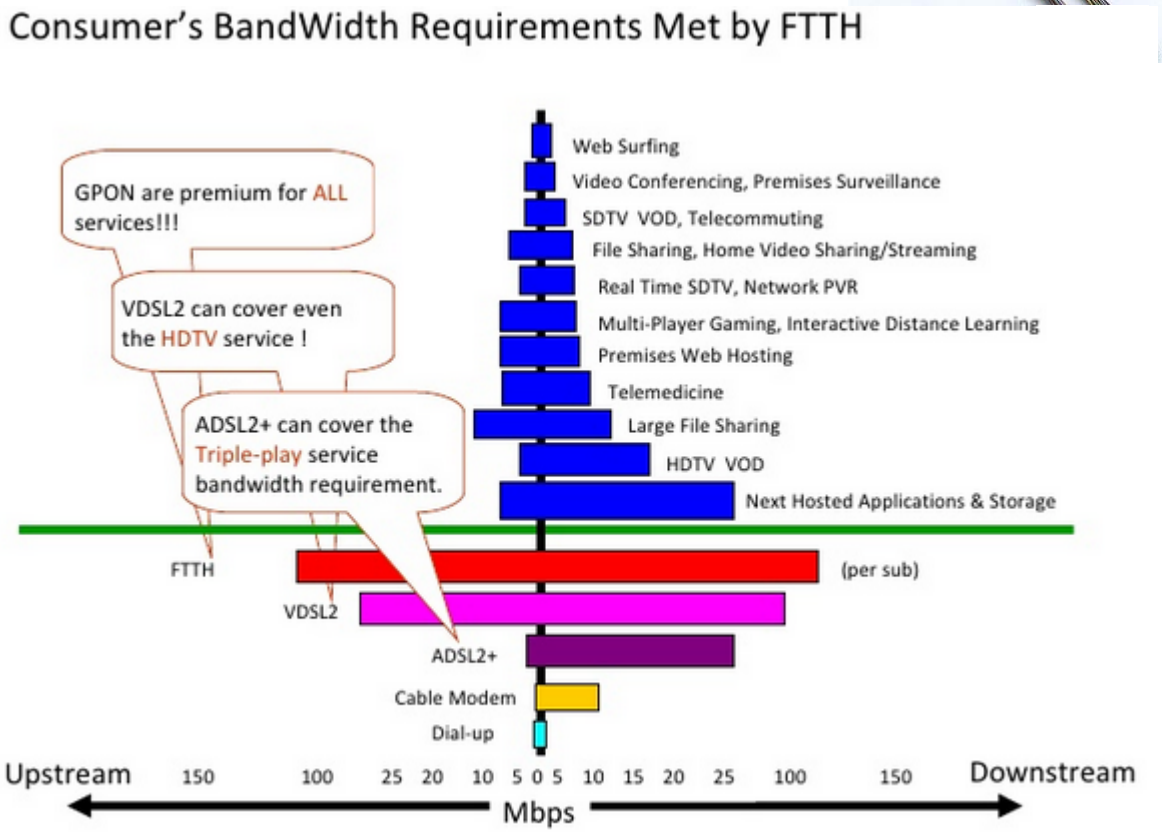
- Passive optical networks face a number of challenges.
- One question is what markets can PONs address.
  - They have mostly been targeted at the residential market so far.
  - The business market does not seem to be a good opportunity.
    - PONs are a shared medium and many businesses will not be happy with service that varies based on the use of other subscribers.
    - Reasonable sized businesses generate and use a lot of data which means that a PON may not be able to serve many subscribers.
    - Point-to-point , non-shared, services are available to business at acceptable prices.

PON Challenges

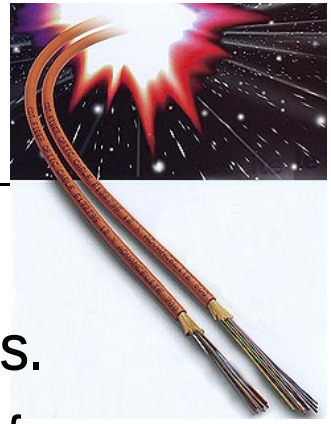
- Next is cost for residential service. Laying fiber to the home is expensive and competition limits the prices that can be charged to the subscriber.
  - The telephone company uses FTTN (neighborhood), which only required laying fiber to the DSLAM – much lower cost than laying fiber to every house.
  - The cable company similarly only had to lay fiber to nodes in their network.
  - Both the telephone company and the cable company were able to use cables to the home that were already depreciated.
  - Verizon, which has FiOS, has announced that they have halted expansion of the service.
- A hybrid fiber/copper approach could be used with G.fast over the copper. Requires fiber very close to the house and access to the copper.

PON Challenges

- Another question is whether the additional bandwidth available with fiber is needed for triple play to the home.
- VDSL and cable systems may provide sufficient bandwidth.



PON Challenges



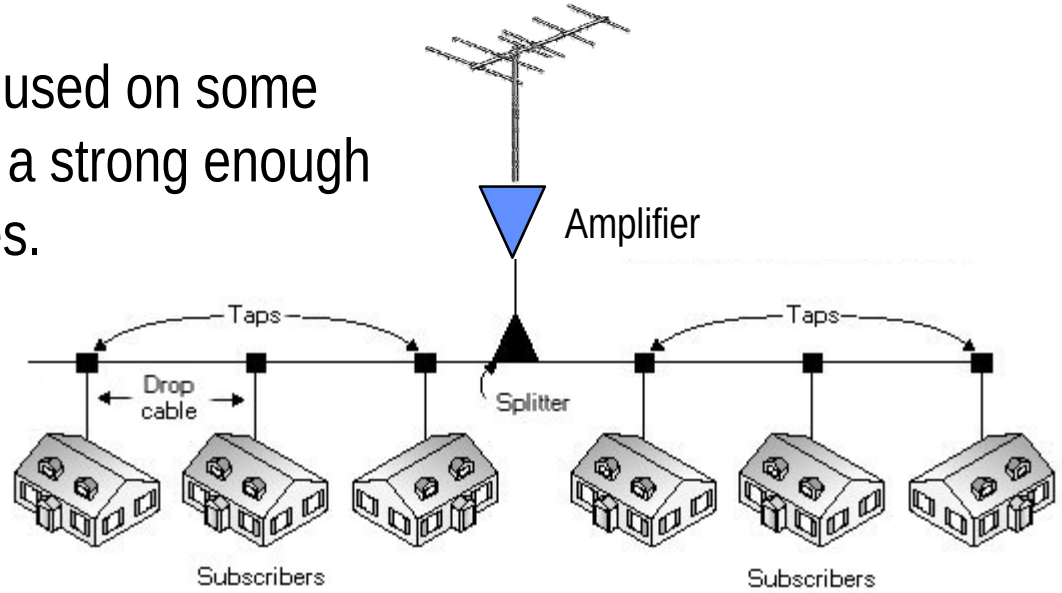
- The next challenge is the changing taste of subscribers.
  - Many subscribers have given up their land lines in favor of cellular phones.
  - Many are also opting out of standard television services in favor of access to video through the Internet.
    - People want to see what they want, when they want to see it, not when the provider gives it to them. This is similar to what happened to music albums.

Hybrid Fiber/Coax (the cable company)

- Cable systems began when someone put up TV antennas on a good site, amplified the signals, and provided the signals to subscribers via coax for a fee.
- The original name was Community Antenna TV (CATV).
- Original systems were all analog, and one way.
- Difficult to manage.
  - Signal level had to be correct at the home or the TV picture had lots of “snow”.
  - The analog amplifiers amplified noise as well as the TV signal.
- But the picture was better than what people could get using their own antenna.

An Early CATV System

- Early systems simply took signals from the air via ordinary TV antennas, amplified them, and distributed the signals to subscribers.
- Signal power management was a major problem.
  - Subscribers close to the amplifier would get too strong a signal.
  - Subscribers far from the amplifier would get too weak a signal and snow.
  - Attenuators had to be used on some houses in order to get a strong enough signal to the far houses.



Early Analog Cable Systems

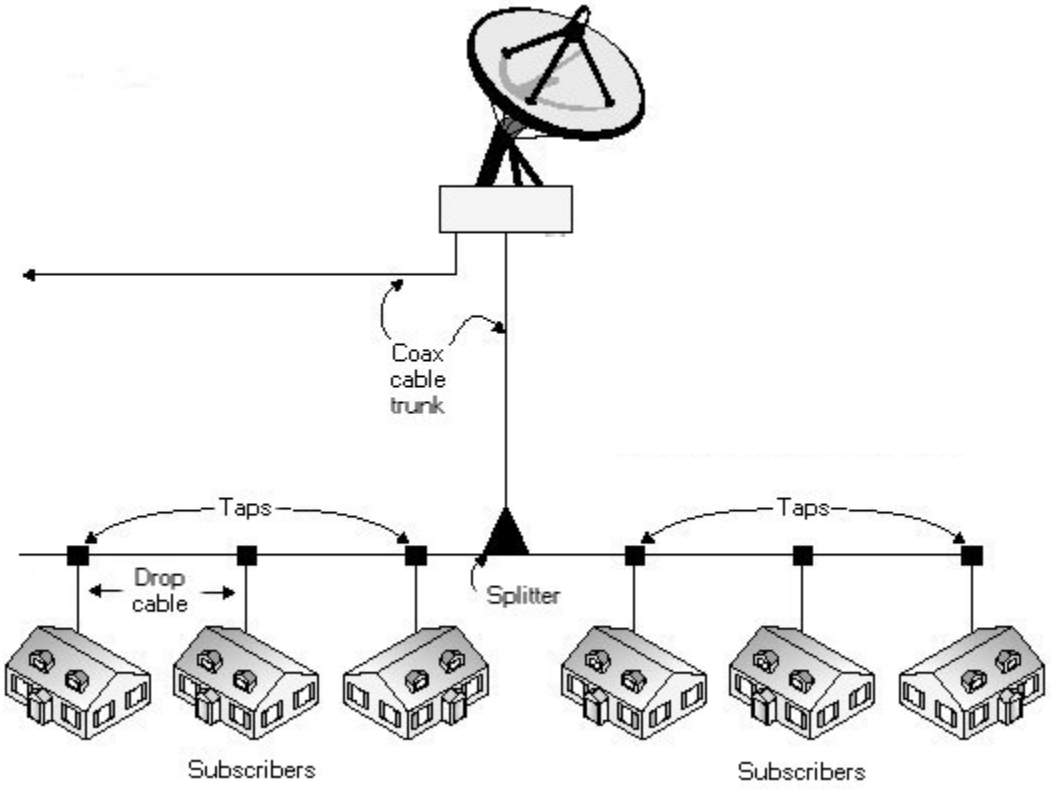
- In the US, the frequency allocations for TV were:
  - Channels 2 – 6, 54MHz to 88MHz.
  - FM stations – 88.1MHz to 107.9MHz
  - Channels 7 – 13, 174MHz to 215MHz.
  - UHF channels were from about 470MHz to 890MHz but cable systems normally only went to 552MHz so any channels above that were translated to available lower slots. UHF channels were sparse.
- These frequencies were carried on the cable so that the standard TV sets could receive the channels, exactly the same as if received over the air.
- This was also an invitation to theft of cable service because no special equipment was required to view the channels.

Early Analog Cable Systems

- The cable companies – also known as Multiple System Operators (MSO) – offered levels of service, with premium channels (e.g., HBO) being an extra charge.
- Problem: How to prevent non-premium subscribers from getting the channels.
  - Originally, they installed notch filters on the line serving the non-premium subscribers to block those channels. People soon learned what was being done and would remove the filters. The cable operator had no way of knowing the filters had been removed.
  - Later, they scrambled the premium channels and required the subscriber to have a set top box (STB) to decode the premium channels. Third parties offered descramblers which would allow viewing premium channels.

Hybrid Fiber Coax

- Over time, operators added additional channels delivered via satellite to their head end. These channels were placed higher in frequency (generally) than the standard channels
- Required a Set Top Box (STB) to translate the frequency to a standard channel (usually channel 2 or 3).
- So the TV was left on Chan 2 or 3 and channel selection was done on the STB.
- System was still all analog and one way.
- Pay per view was ordered via a low speed modem.

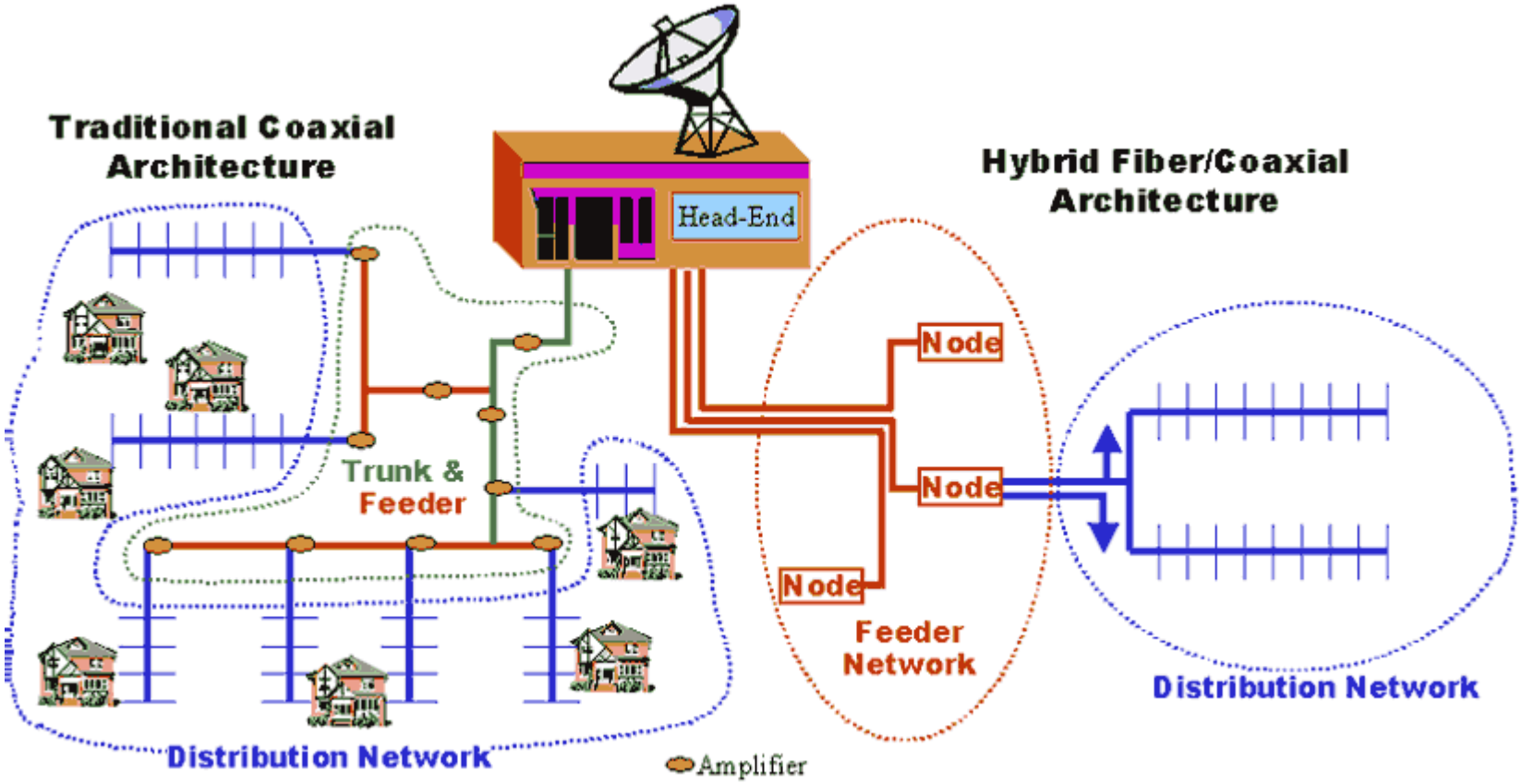


The Big Upgrade

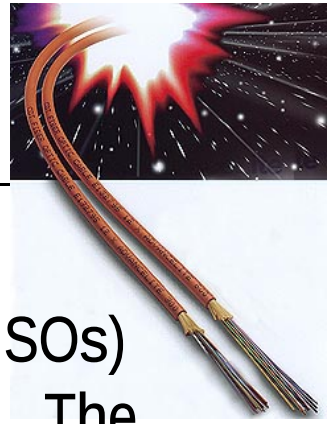
- By the early 1990s cable operators recognized that their systems needed to be upgraded to two way.
  - Allowed easy ordering of Pay Per View (PPV).
  - Set top boxes could be remotely activated, enabled for additional services, and monitored.
- As part of the upgrade, most operators also upgraded the system to 750MHz or 1GHz.
  - Amplifiers and splitters.
- Fiber replaced trunk coax cables.

Example of Upgraded 2-way CATV

- This figure contrasts the original coaxial distribution system with the upgraded fiber/coax system.



Upgraded Systems



- Along with the facility upgrades, the cable operators (MSOs) began transmitting television channels in digital format. The only way those channels could be received was with an STB.
- TV was encoded in MPEG while the sound was in MP-3.
- For a while, they continued to offer the basic channels in analog form so that customers who did not use STBs could receive them. But gradually, they forced everyone to STBs and made all the channels digital.
- Digital was more bandwidth efficient which allowed the MSOs to either offer more channels, or to use the bandwidth to provision telephony and Internet access.

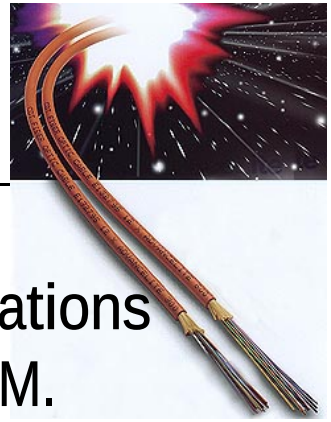
Upgraded Systems

- The conversion to digital essentially eliminated piracy of cable service.
- The signals could not be (easily) decoded without an STB.
- The two way system allowed them to recognize STBs and authorize them.
- It's technically possible to "hack" a cable system but it would require someone with a lot of technical knowledge and resources.

Data over Cable

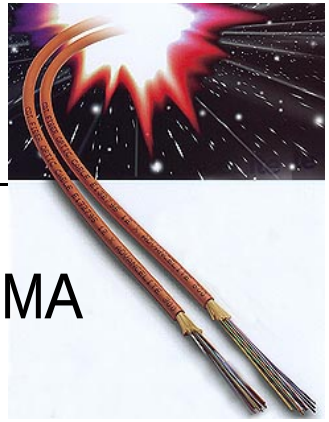
- Once the Internet took off, the cable system operators recognized that they could provide Internet access across their systems.
- The formed a group that produced the Data Over Cable Service Interface Specification (DOCSIS) which defined cable modem operation.
- DOCSIS has gone through several iterations:
  - DOCSIS 1.0, released 1997.
  - DOCSIS 1.1, released 1999.
  - DOCSIS 2.0, released 2001.
  - DOCSIS 3.0, released 2006.
  - DOCSIS 3.1, released 2013.

DOCSIS



- For DOCSIS 1.0 and 1.1, the upstream bandwidth allocations can go from 200KHz to 3200KHz, using QPSK or 16QAM. Rates are from about 400Kbps to 13Mbps.
- Downstream bandwidth allocations are a bit over 6MHz, using 64QAM or 256QAM, giving rates of 36 and 43Mbps.
- DOCSIS 2.0 increased the possible upstream bandwidth to 6400KHz, and the modulation to 128QPSK (max), giving rates from 400Kbps to 36Mbps.
- DOCSIS 3.0 allowed n x (200 – 6400KHz) concatenated blocks of bandwidth, giving a max rate of n x 36Mbps.
- Downstream can be n x 6MHz channels giving a max rate of n x 43Mbps.

Basic DOCSIS Modem Operation



- When the cable modem (CM) is turned on, it uses a CSMA channel to register itself to the CMTS (cable modem termination system).
- The CM and the CMTS will negotiate an encryption key for data sent to and from that modem.
- After that, the upstream is via TDM. The CMTS will send a frame to discover which CMs have data to be sent. It will also contain a map indicating when the CM can reply.
- The CMTS will next provide a MAP frame which provides time slot grants for when each CM can send data.
- Voice is automatically granted a slot every 10 to 20ms until the call ends.

Future of Hybrid Fiber/Coax

- DOCSIS 3.1 offers the ability to put much higher rates across the system, using OFDM technology.
- As fiber is brought closer to the home, the coax to the home has the ability to handle these rates much better than twisted pair – and it's already there!
- The cable company may have a cost advantage in delivering very high Internet access when compared to the telephone company and VDSL or passive optical networks.

Ethernet

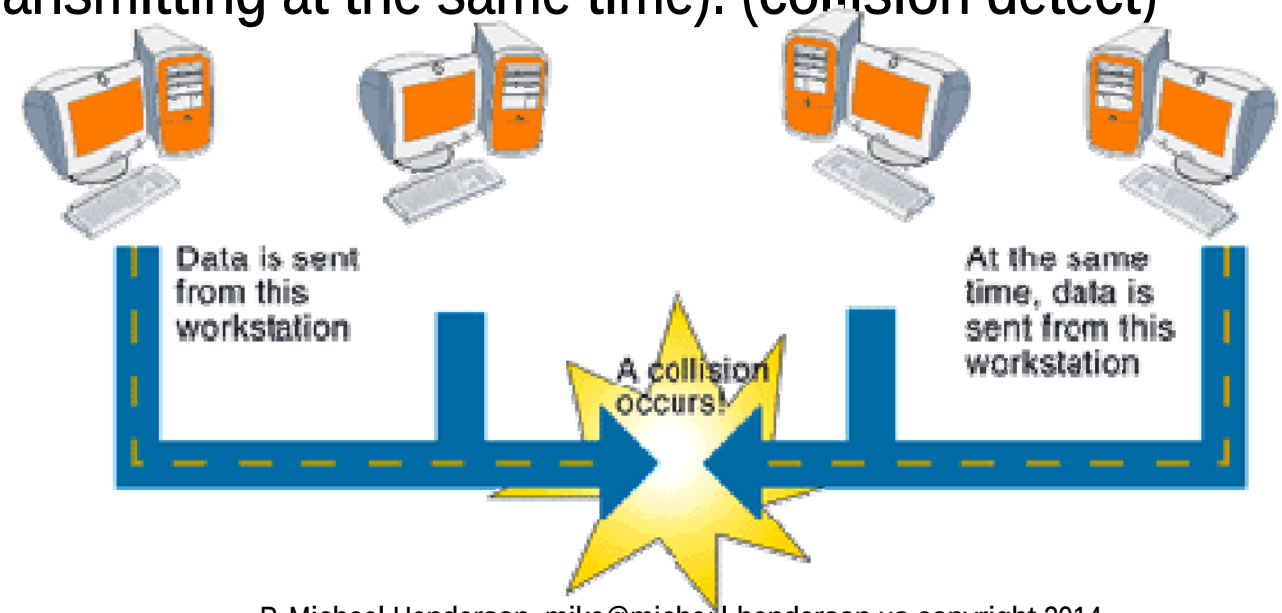
- We need to look at Ethernet at this time because its frame may be carried by many of the systems we will examine later.
- The description offered here will not go into all the details of the system but will only offer an overview.
- We'll examine Ethernet in pieces:
  - The Media Access Control (MAC) protocol. We'll look at CSMA-CD even though it's not used much in Ethernet but is used in other applications.
  - The Ethernet frame.
  - The physical medium. We'll mostly look at twisted pair (e.g., Cat-5 cable).
- WARNING – Ethernet transmits octets Least Significant Bit (LSB) first. This is counterintuitive and can cause problems in interpretation of the data actually transmitted.

Ethernet

- Ethernet is a Layer 2 protocol. It does not guarantee delivery of any packets sent to it for delivery.
- If errors are detected in received frames, the frames are discarded without notice to any higher level protocols.
- If frames cannot be sent because of congestion or other reasons, the frames will be discarded without notice to higher level protocols
- Ethernet is a “best efforts” system.

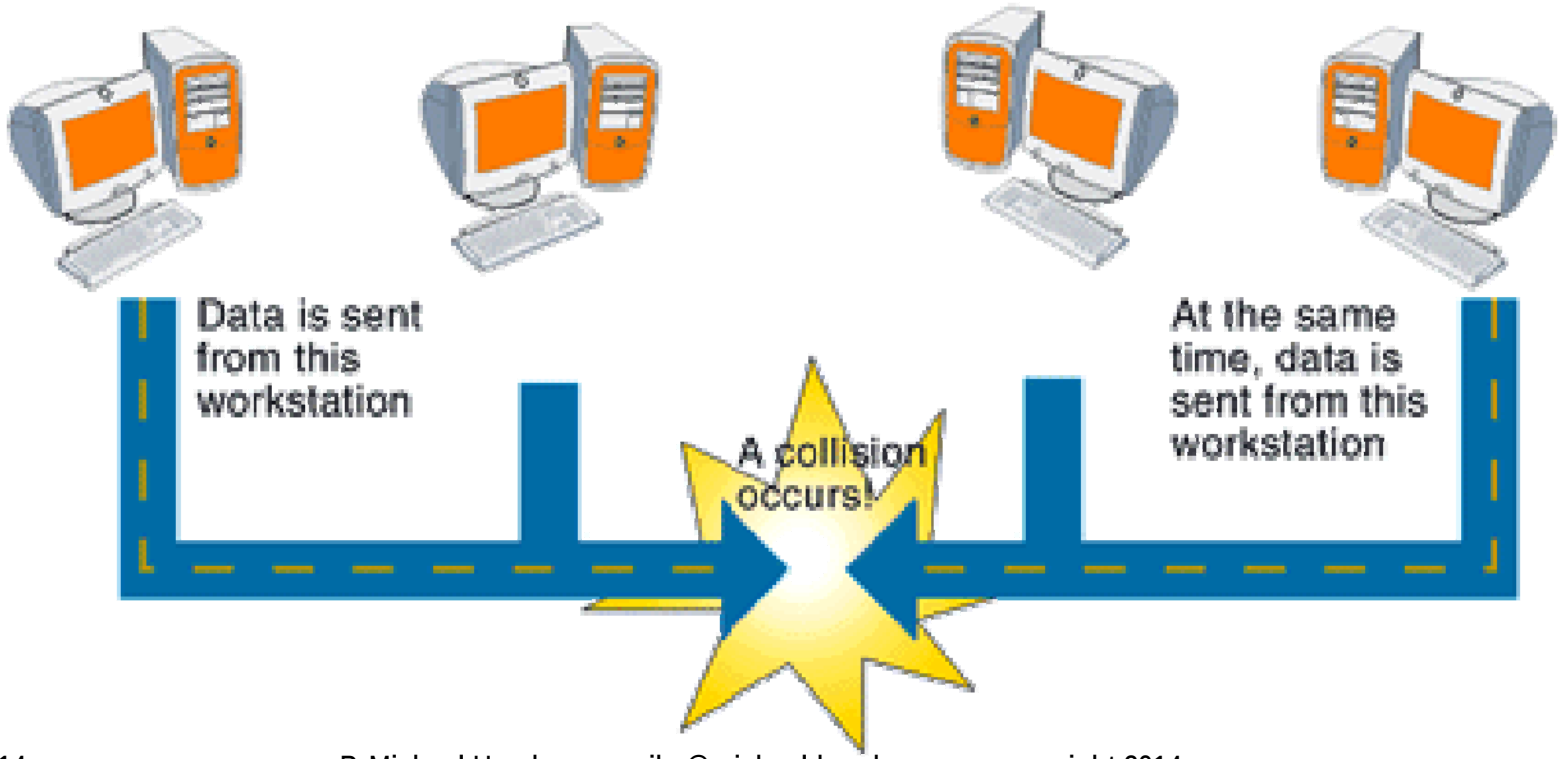
Ethernet – CSMA-CD

- Originally, Ethernet was designed to share a transmission channel. (multiple access)
- A station would listen (carrier sense) to see if there was activity on the channel, and if not, would transmit.
- While transmitting, the sending station would listen to determine if a collision occurred (another station was transmitting at the same time). (collision detect)



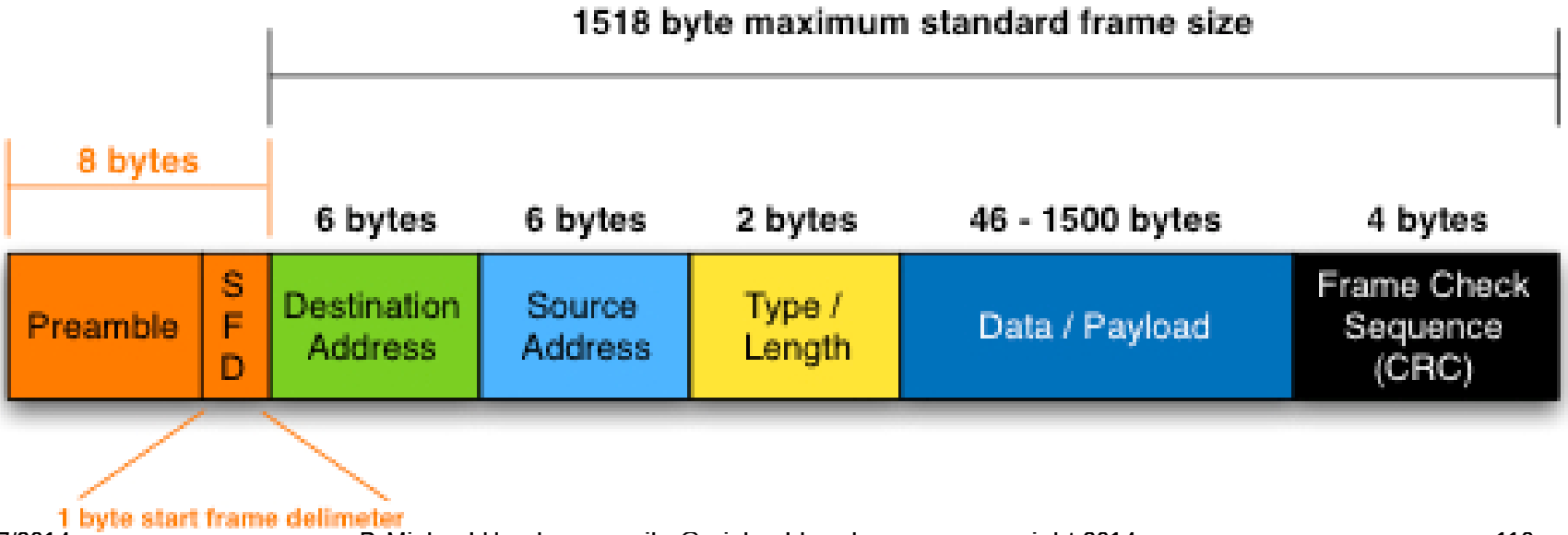
Ethernet – CSMA-CD

- If a collision occurred, the station would send the jamming signal to make sure the other station knew of the collision.
- Then wait a random time period (the back-off time) and go through the process again.



Ethernet Frame

- We're only going to look at the 802.3 frame.
- The preamble is 7 octets of 1010 1010 (alternating 1s and 0s).
- Then a Start Frame Delimiter of 1010 1011.
- Next is the Destination and Source MAC addresses, each 48 bits long (6 octets).

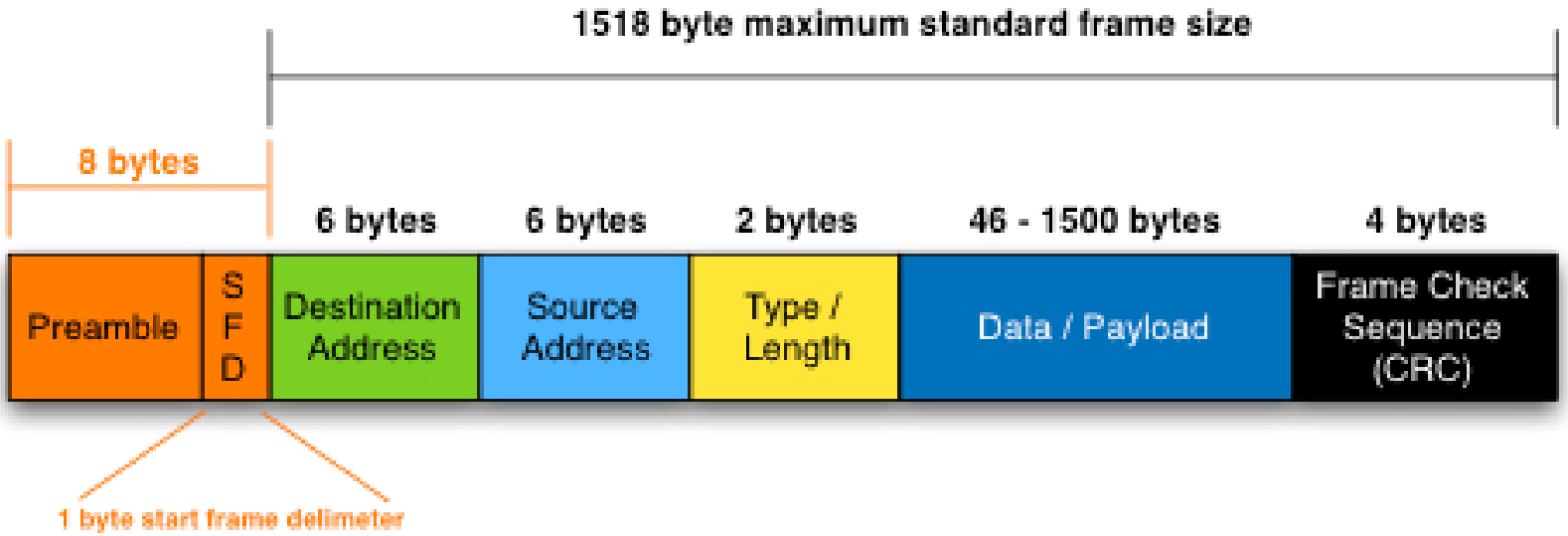


MAC Addresses

- Ethernet MAC addresses are globally unique. That is, every Ethernet Network Interface Controller (NIC) has a unique address.
- With 48 bits, we can have over 280 trillion addresses. Each vendor is assigned a 24 bit address, called the Organizational Unique Identifier (OUI), which becomes the first 24 bits in the address. The last 24 bits are sequentially assigned by the vendor. So each vendor can manufacture over 16 million units before asking for an additional OIU.
- A few addresses are reserved:
  - All 1s is the broadcast address.
  - If the least significant bit in the first octet is a 1, the address is a multicast address. (This will be the first bit transmitted of the first octet of the address.)

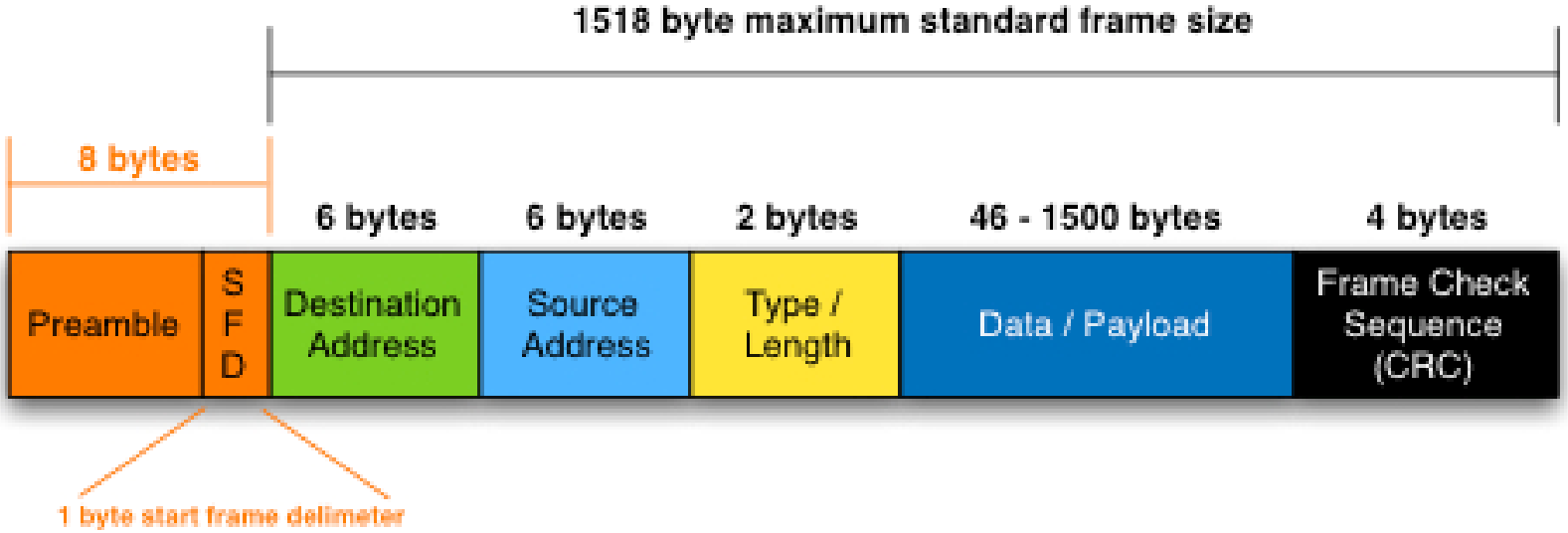
Ethernet Frame

- The length field is used two ways:
  - To indicates the number of data octets (including the LLC) in the data field. If less than 46 the data must be padded to make 46. The maximum size is 1500.
  - If the value of the length field is 0x0600 or greater, the field is being used to indicate the protocol being carried, and the protocol will indicate the length. Some common protocol indicators are 0x0800 for IPv4, 0x0806 for ARP, 0x86DD for IPv6.



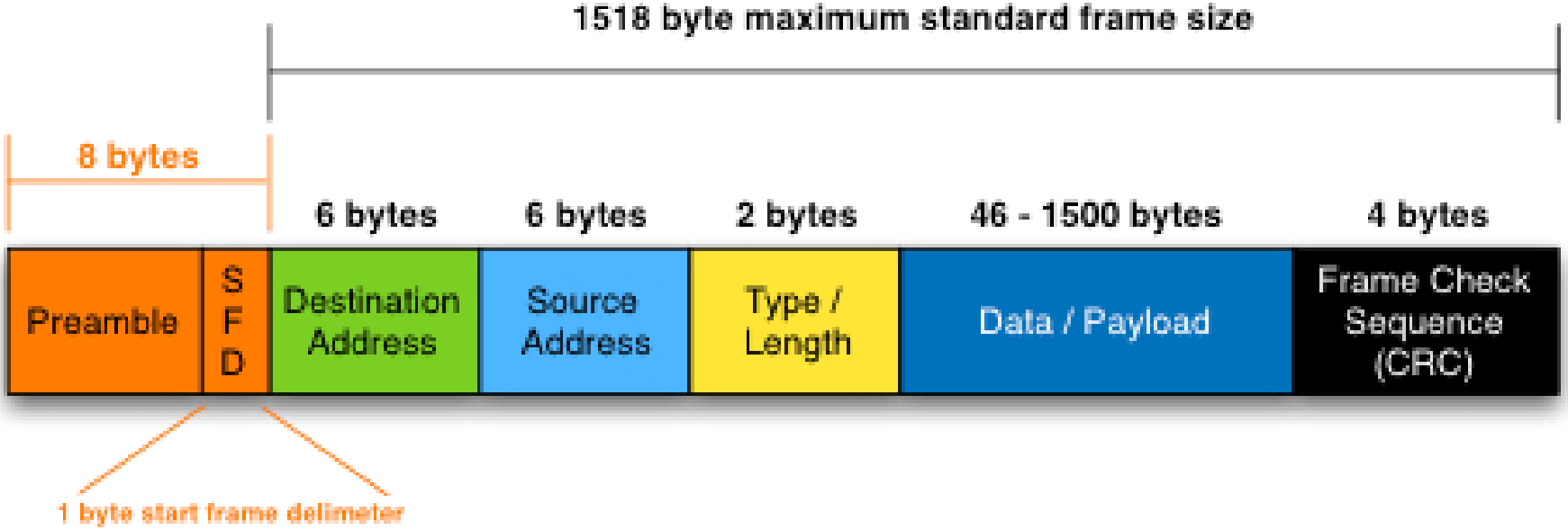
Ethernet Frame

- A question arises when the “Type/Length” field is used to indicate the protocol carried. “How do you find the end of the frame?”
- For 100BASE-TX, a special delimiter is sent – 01101 00111 – after the Frame Check Sequence (FCS). Note that this is two 5 bit characters. 100 BASE-TX converts nibbles to 5 bit characters using a technique called 4B5B which we’ll discuss later. All of the other versions have a similar End of Stream Delimiter (ESD) (except 10BASE-T).



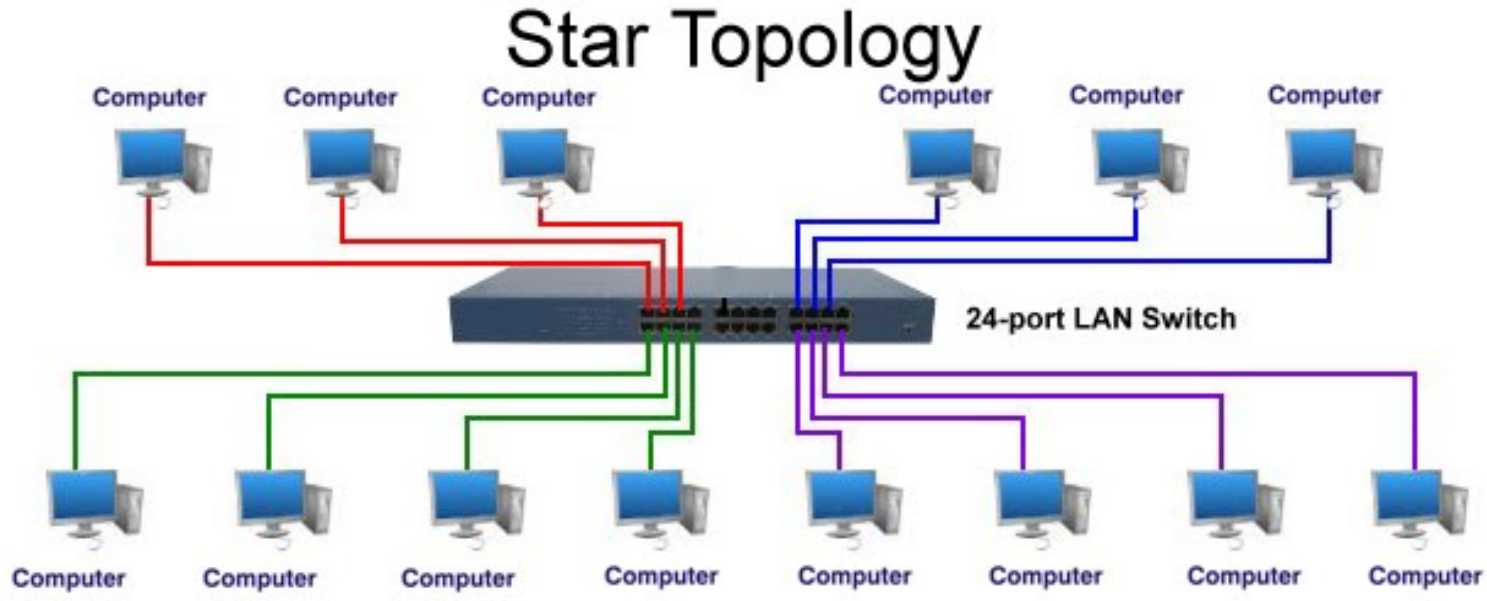
Ethernet Frame

- The data field can be anywhere from 46 octets to 1500 octets. If it is not at least 46 octets (making the overall frame 64 octets), the field is padded to make it 46 octets.
- The last field is the Frame Check Sequence, which is a 32 bit CRC.
- Just FYI, the idle character is 11111 – five 1s – sent continuously when there's no data to be sent.



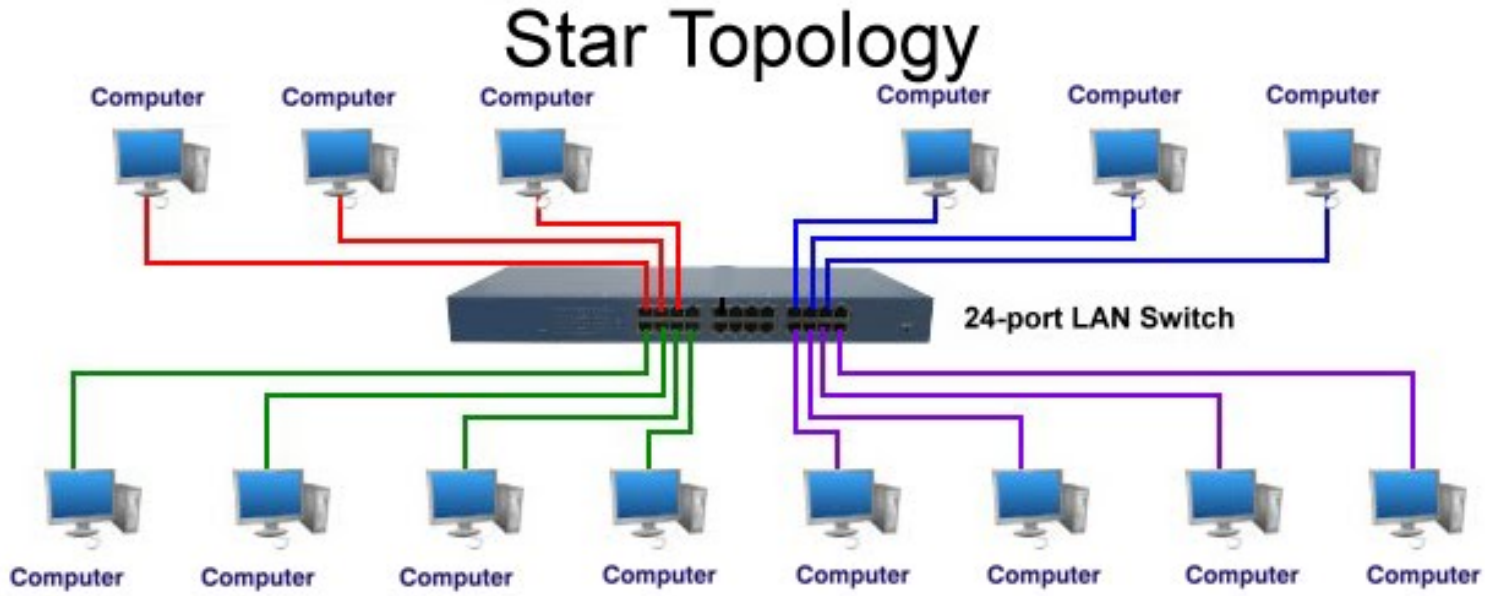
Media Access Control Protocol

- Today, Ethernet operates as a star topology and the links are almost always full duplex.
- The user's computer connects to an Ethernet switch which has buffer memory and functions somewhat like a router.

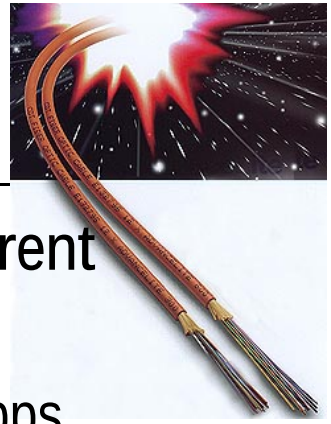


Ethernet Switch

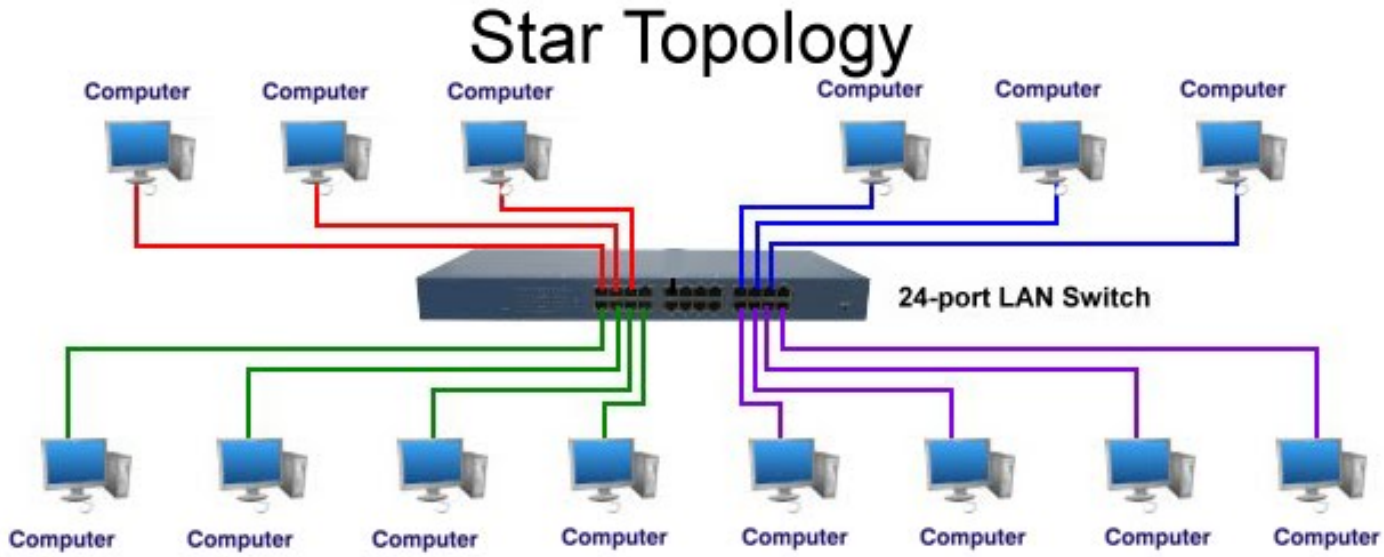
- The switch “learns” the network by observing the source address field of the frames.
  - That is, it watches frames and remembers the source addresses. When a frame comes to it with that address in the destination address, it sends the frame out on that one port.
  - If it does not have the address, it sends a received frame out on all ports.



Ethernet Switch



- Note that different connections to a switch may be different speeds.
  - A user may have an Ethernet card that only supports 10Mbps.
  - A cable may have a problem causing a link to fall back to a lower speed.
- This means that the switch must buffer each frame before transmitting it.



Media Access Control Protocol

- The description of the Ethernet switch I gave begs a question: It's very good that one Ethernet unit can contact another, but we don't route on Ethernet addresses – we route on IP addresses. How do we get from an IP address to an Ethernet address so we can make a connection?
- The answer is ARP (Address Resolution Protocol). When we discuss TCP/IP we'll look at this protocol in more detail to see how the higher levels of the protocol stack can use Ethernet addresses to make connections.

Ethernet Physical Media

- The description of the media is somewhat standardized but has changed over time.
- First comes the rate, such as 10, 100, 1000, etc.
- Then the word BASE meaning a baseband signal.
- Then a hyphen.
- Next, there's several options: T means twisted pair, X means that block coding (such as 4B5B) is used, F is fiber. In the higher speeds, C means copper. The standards body sort of jumped around in this area.

Ethernet Twisted Pair

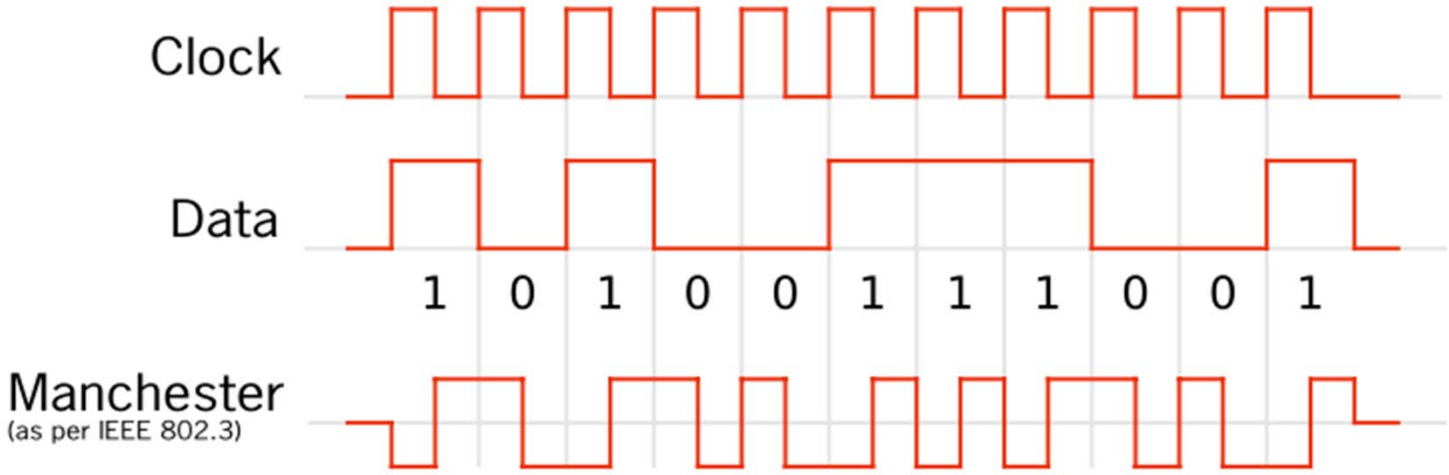
- All of the specifications for twisted pair for Ethernet are for maximum lengths of 100 meters.
- This is broken into 90 meters for the run and 10 meters for the connection to the PC.
- The 100 meters is a limit for two reasons:
  - In half-duplex operation, it's necessary for proper operation of the CSMA-CD (signal propagation time).
  - The encoding and power into the line is specified for the maximum losses at 100meters.
- For full-duplex operation you could go more than 100meters by using higher grade cable.

10BASE-T

- Uses two twisted pair, one Tx and one Rx, at Cat-3 level (telephone wire).
- Maximum length of 100meters.
- Uses Manchester encoding, same as for the coax version of Ethernet. 0 is high-to-low transition and 1 is low-to-high. Note: No scrambler or block encoding is needed because of the Manchester encoding.
- Bandwidth is 20MHz because of the Manchester encoding.
- Very similar to operation on the coax “Ethernet”.
- Was originally defined only for half-duplex operation but most vendors added full-duplex later.

Manchester Coding

- Manchester coding is a type of phase encoding.
- Each bit has at least one transition, and it is DC balanced.
- A Manchester code insures frequent line voltage transitions and thus provides good clocking.
- But it takes a lot of bandwidth.
- Note: 0 is high to low, 1 is low to high.



10BASE over Fiber

- Several specifications developed
  - 10BASE-FL
  - 10BASE-FB
  - 10BASE-FP
- 10BASE-FL was perhaps the only version to get any significant use.
  - Full duplex operation over two mulitmode fibers, 2km.

4B5B Encoding

- Here's the Ethernet 4B5B encoding table. Note the high density of 1s in the output words.

Input word	Output word	Other output words	
0 0 0 0	1 1 1 1 0	1 1 1 1 1	Idle symbol
0 0 0 1	0 1 0 0 1	0 0 1 0 0	Halt line symbol
0 0 1 0	1 0 1 0 0	1 1 0 0 0	Start symbol
0 0 1 1	1 0 1 0 1	1 0 0 0 1	Start symbol
0 1 0 0	0 1 0 1 0	0 1 1 0 1	End symbol
0 1 0 1	0 1 0 1 1	0 0 1 1 1	Reset symbol
0 1 1 0	0 1 1 1 0	1 1 0 0 1	Set Symbol
0 1 1 1	0 1 1 1 1	0 0 0 0 0	Invalid
1 0 0 0	1 0 0 1 0	0 0 0 0 1	Invalid
1 0 0 1	1 0 0 1 1	0 0 0 1 0	Invalid
1 0 1 0	1 0 1 1 0	0 0 0 1 1	Invalid
1 0 1 1	1 0 1 1 1	0 0 1 0 1	Invalid
1 1 0 0	1 1 0 1 0	0 0 1 1 0	Invalid
1 1 0 1	1 1 0 1 1	0 1 0 0 0	Invalid
1 1 1 0	1 1 1 0 0	0 1 1 0 0	Invalid
1 1 1 1	1 1 1 0 1	1 0 0 0 0	Invalid

100BASE over Fiber

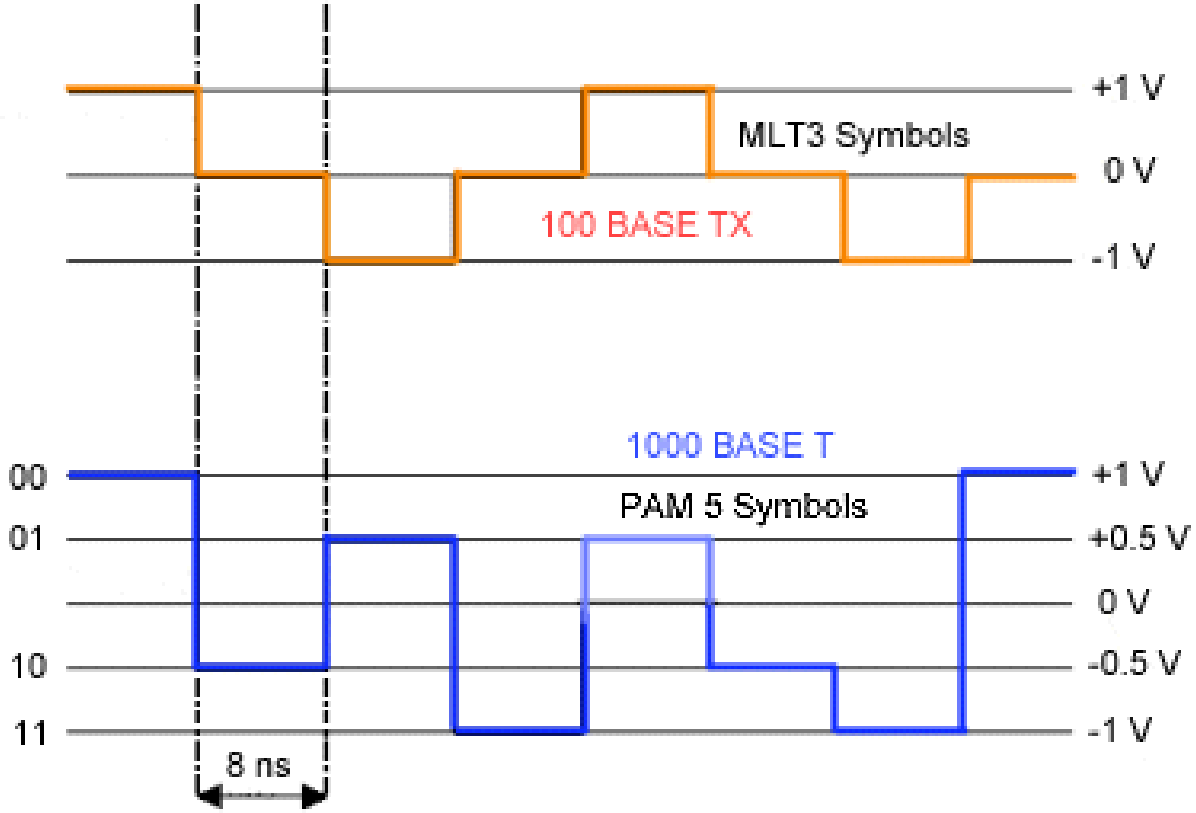
- Several specifications for 100BASE over fiber were developed:
  - 100BASE-FX - up to 2Km over 2 multimode fibers, 1310nm.
  - 100BASE-SX – Similar to FX, but at 850nm. Industry standard, not IEEE standard.
  - 100BASE-BX – Full duplex over one single-mode fiber, one side transmitting at 1310nm and the other at 1550nm. Up to 40Km.
  - 100BASE-LX – Two single-mode fibers, 1310nm, 10Km.
- All used the 4B5B encoding for 1s density.

1000BASE-T

- Uses four twisted pair of Cat 5 or better (Cat-5e recommended), full duplex, using echo and self-NEXT cancellation. Each pair communicates 250Mbps, full duplex.
- Maximum length of 100meters.
- Frame is scrambled prior to transmission for 1s density, different scrambler for master and slave. Master/slave is determined at auto-negotiation.
- Uses 5 level PAM at 125Mbaud.
- Bandwidth is 62.5MHz, since it takes two symbols to make a Hz.
- Also uses a 4D 8 state trellis code for error correction.

MLT-3 and PAM-5

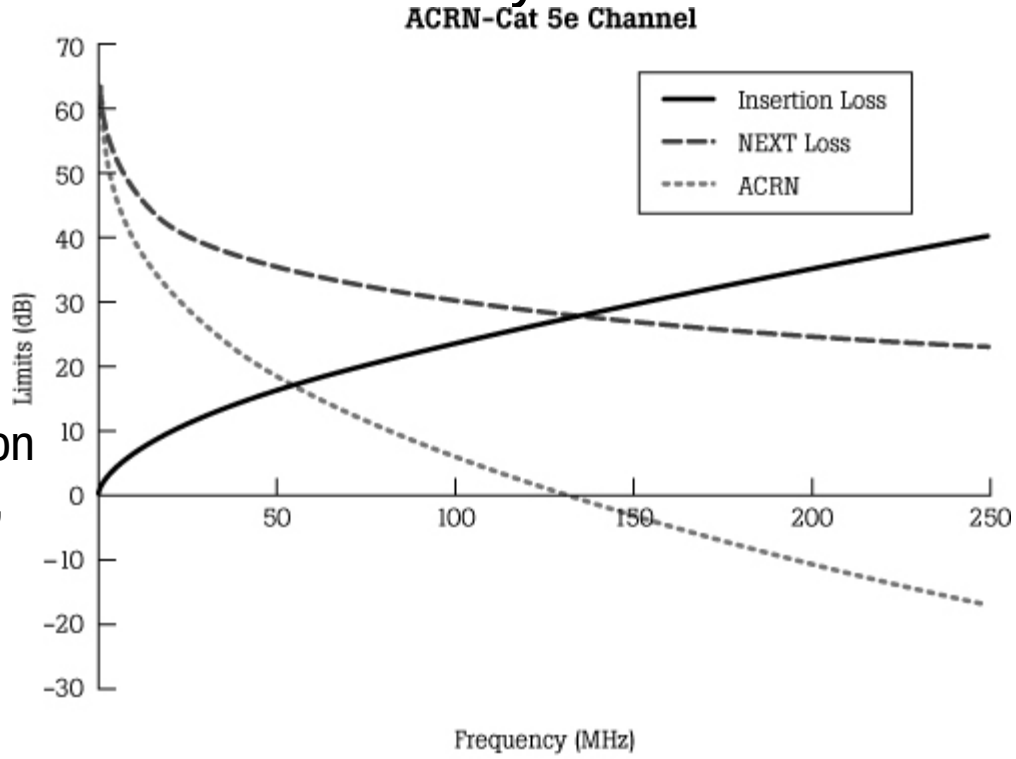
- Here's an example of the line coding for 100BASE-TX and 1000BASE-T. MLT-3 and PAM-5 signaling.



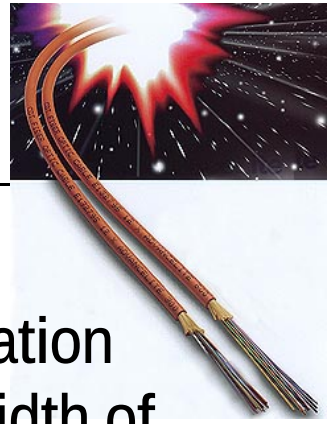
Bandwidth and ACR of Cat-5e

- Note that the attenuation and NEXT cross close to 125MHz.
- There is more than NEXT that affects the bandwidth so the actual bandwidth is less than shown.
- ACRN is the difference, in dB, between Attenuation and NEXT. Note that it is zero where they cross.

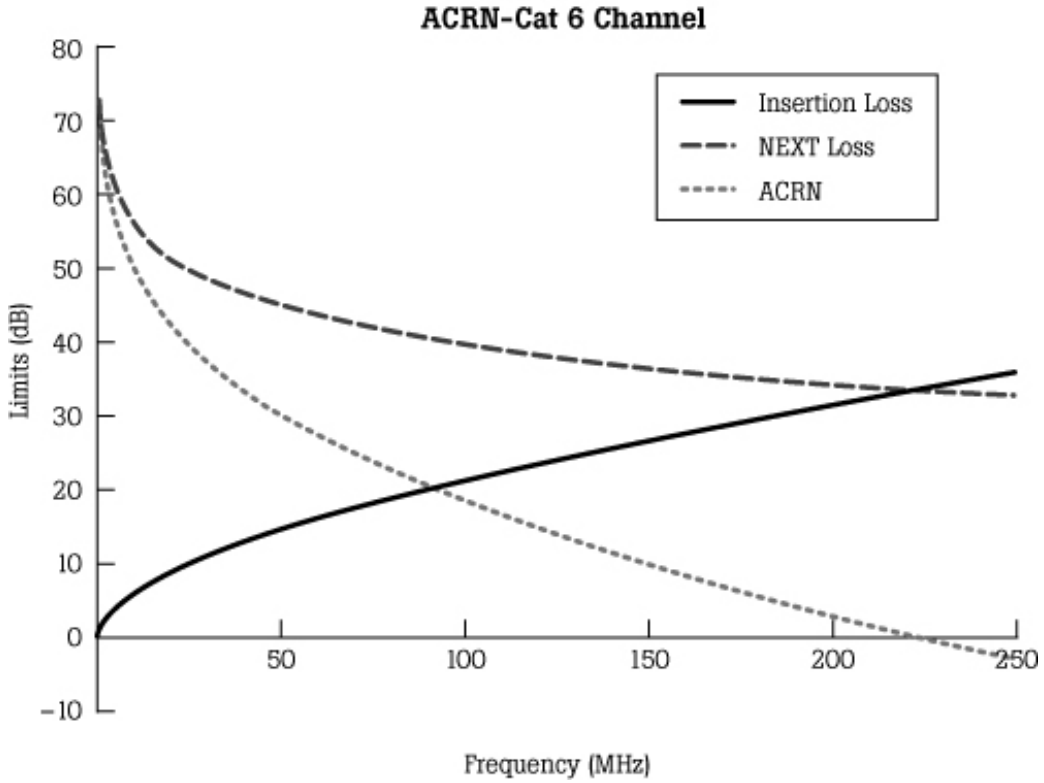
ACRN= attenuation to crosstalk ratio, near end



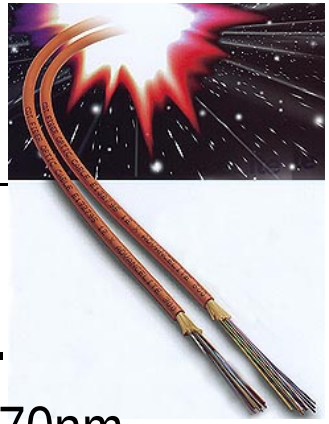
Category 6 cable



- Category 6 cable gives a slight improvement in attenuation but a significant gain in NEXT. This provides a bandwidth of about 200MHz.

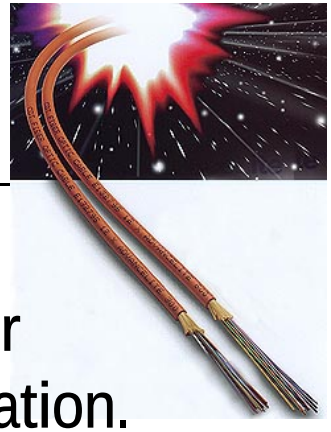


1000BASE over Fiber



- A significant number of standards were developed for 1000BASE operation over fiber. All use 8B10B coding.
  - 1000BASE-SX – Full duplex over two multimode fibers, 770nm to 860nm, 550meters.
  - 1000BASE-LX – Full duplex over two single mode fibers, 1,270nm–1,355nm, 10Km.
  - 1000BASE-LX10 – Similar to LX but up to 10km.
  - 1000BASE-EX – Similar to LX10 but will go 40km. Industry standard.
  - 1000BASE-BX10 – Full duplex over one single-mode fiber, 1310nm and 1490nm, 10km.
  - 1000BASE-ZX – Full duplex over two single-mode fibers, 1550nm, 70km. Industry standard.

10GBASE-T



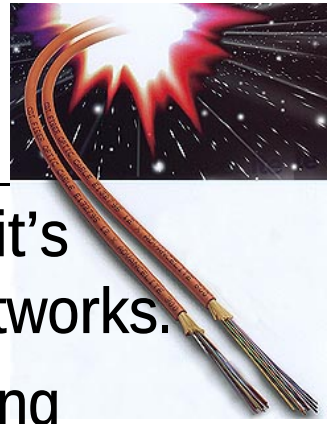
- Uses four twisted pair of Cat 6 (screened to 500MHz) or better, full duplex, using echo and NEXT/FEXT cancellation. Each pair communicates 2.5Gbps, full duplex.
- Maximum length of 100meters.
- Frame is scrambled prior to transmission for 1s density.
- Uses 16 level PAM at 800Mbaud.
- Bandwidth is 400MHz, since it takes two symbols to make one Hz.
- Uses a LDPC code for error correction plus some complex constellation mapping.
- Goal of a BER of 10^{-12} at 100m.

10GBASE over Fiber

- A significant number of standards were developed for 10GBASE operation over fiber (all are full-duplex):
  - 10GBASE-SR – Multimode, 850nm, 400meters.
  - 10GBASE-LR – Single-mode, 1310nm, 10km.
  - 10GBASE-ER – Single-mode, 1550nm, 40km.
  - 10GBASE-ZR – Single-mode, 1550nm, 80km, industry standard.
  - 10GBASE-LRM – Mulitmode, 1310nm, 220m.
  - 10GBASE-PR – Single-mode, 1270 nm/1577 nm, 20km, for passive optical networks (GPON).

Networks and Protocols

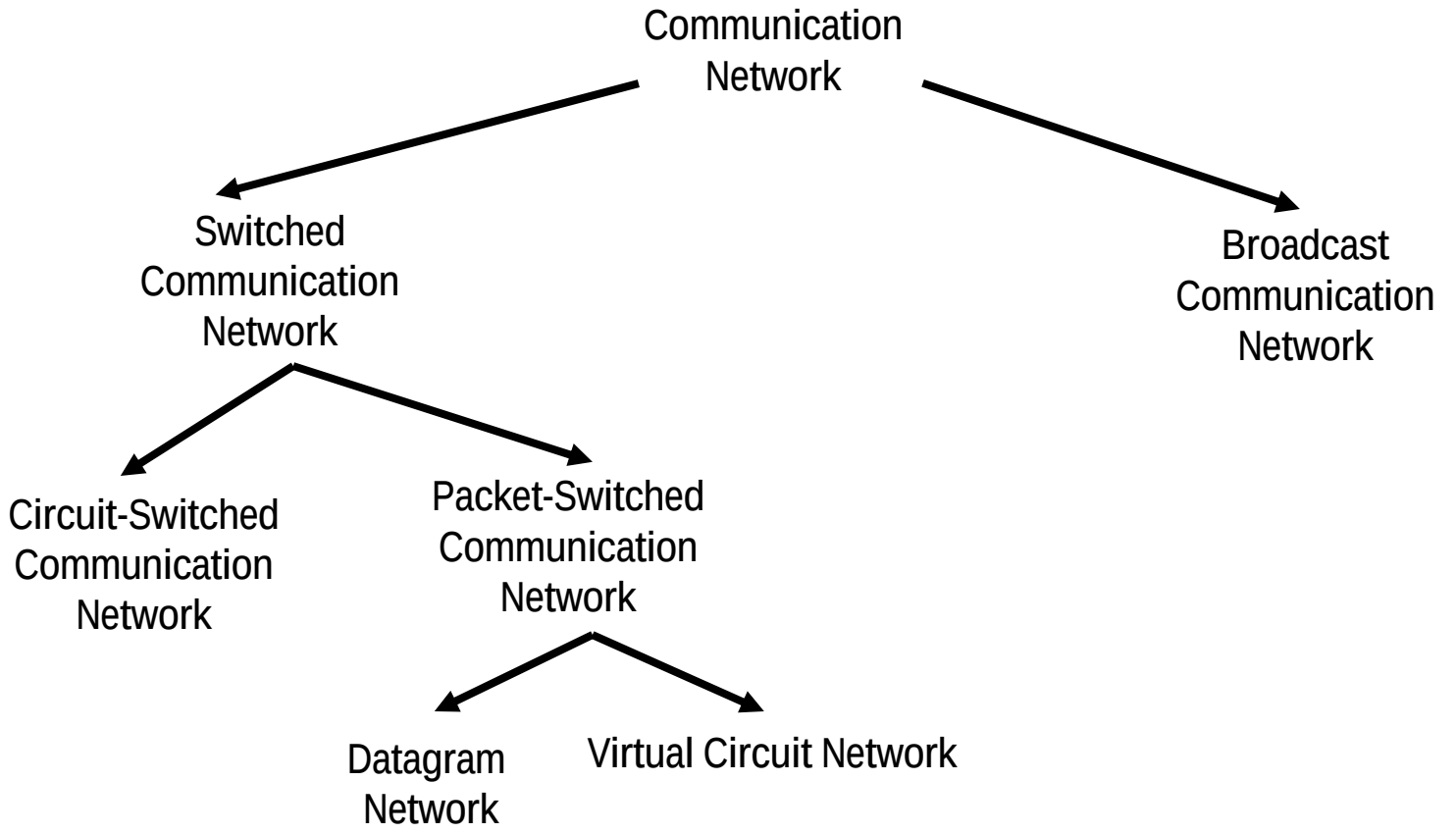
Networks and Protocols



- We've looked at most of the building blocks and now it's time to begin looking at how data is transported in networks.
- There are basically two ways to connect communicating devices: (1) With dedicated communications lines, or (2) by switching/routing.
- Early networks used dedicated lines. Example: Automated Teller Machines were connected to the bank's computer over dedicated leased telephone lines, with modems on each end. These networks are expensive and wasteful.
- Switched/routed networks make better use of communications bandwidth but require more "intelligence" in the network.
- Some networks use the concept of fixed circuits but implement them through switching.

A Taxonomy of Network Communications

- Networks can be classified by how they exchange information.

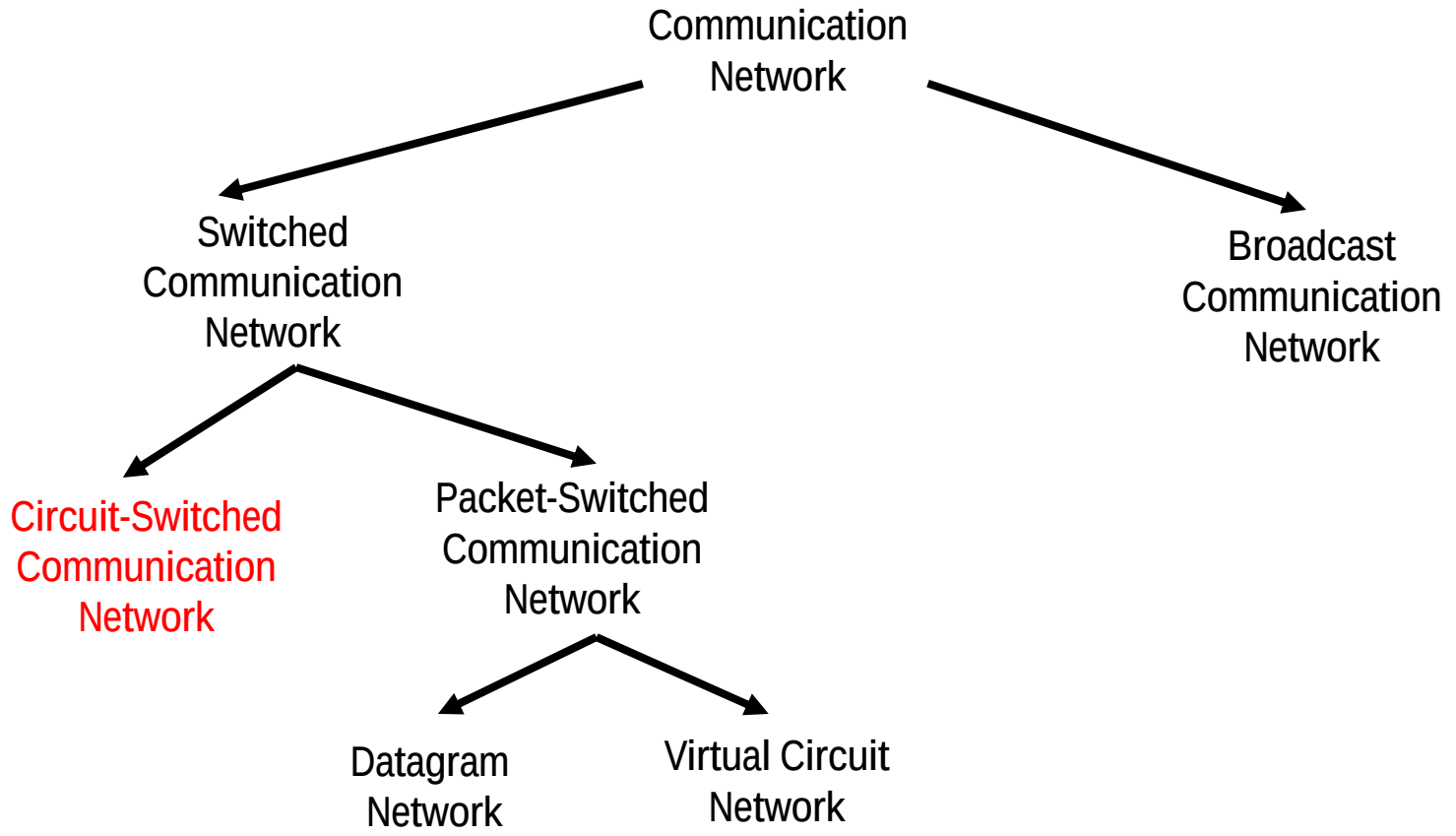


Broadcast vs Switched

- On broadcast networks, the information sent by one node goes to every other node.
  - Example: Ethernet.
- The problem for broadcast networks is to manage access to the transmission medium.
- On switched networks, information is only sent to the selected nodes.
  - Example: the telephone system.
- The problem for switched networks is routing to the intended recipient.

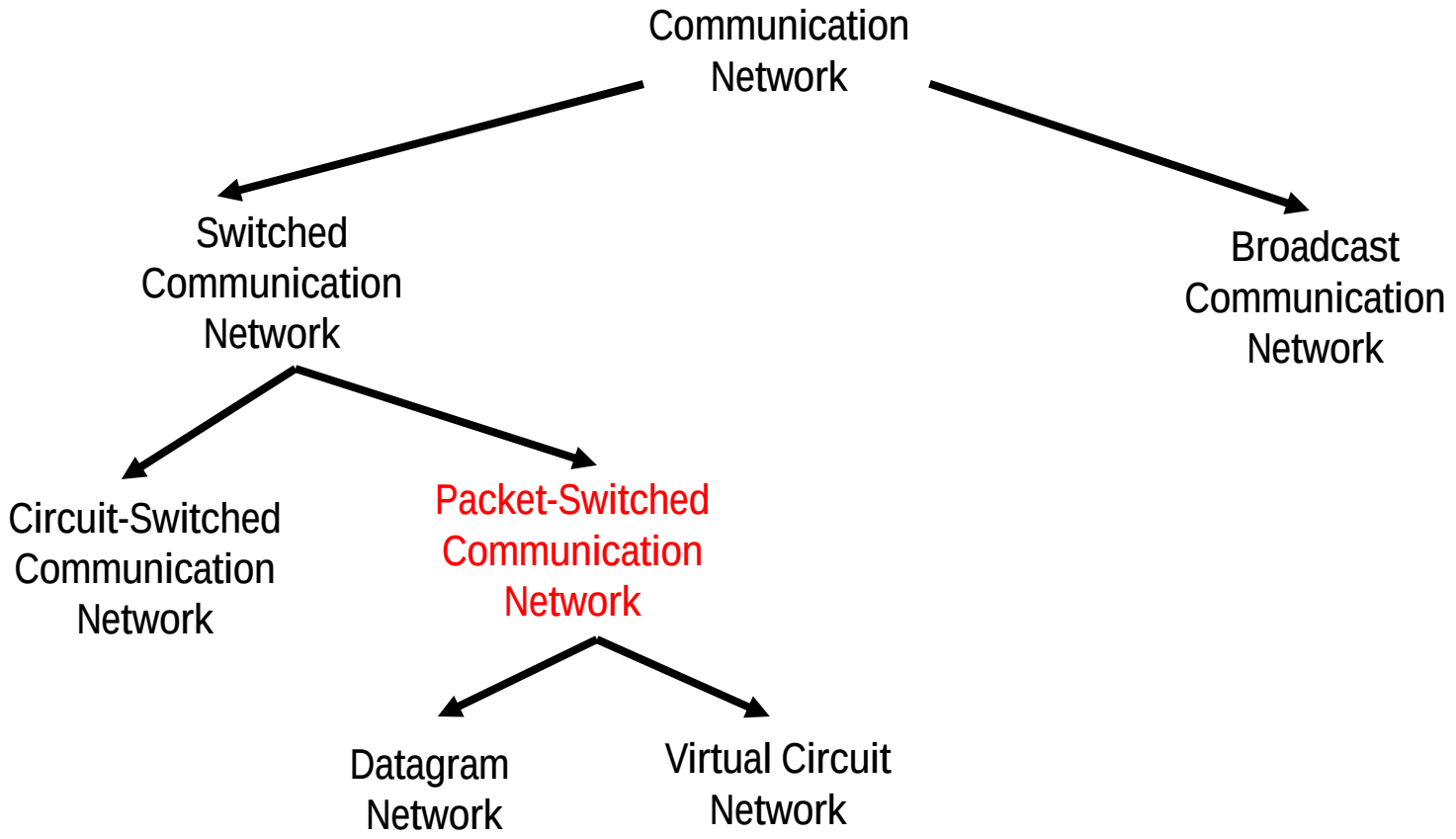
Circuit Switching

- We looked at circuit switching when we covered voice communications so I won't spend any time on it here.

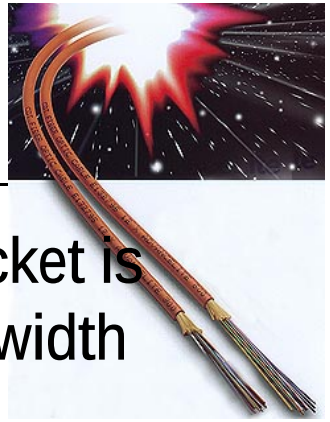


Packet Switching

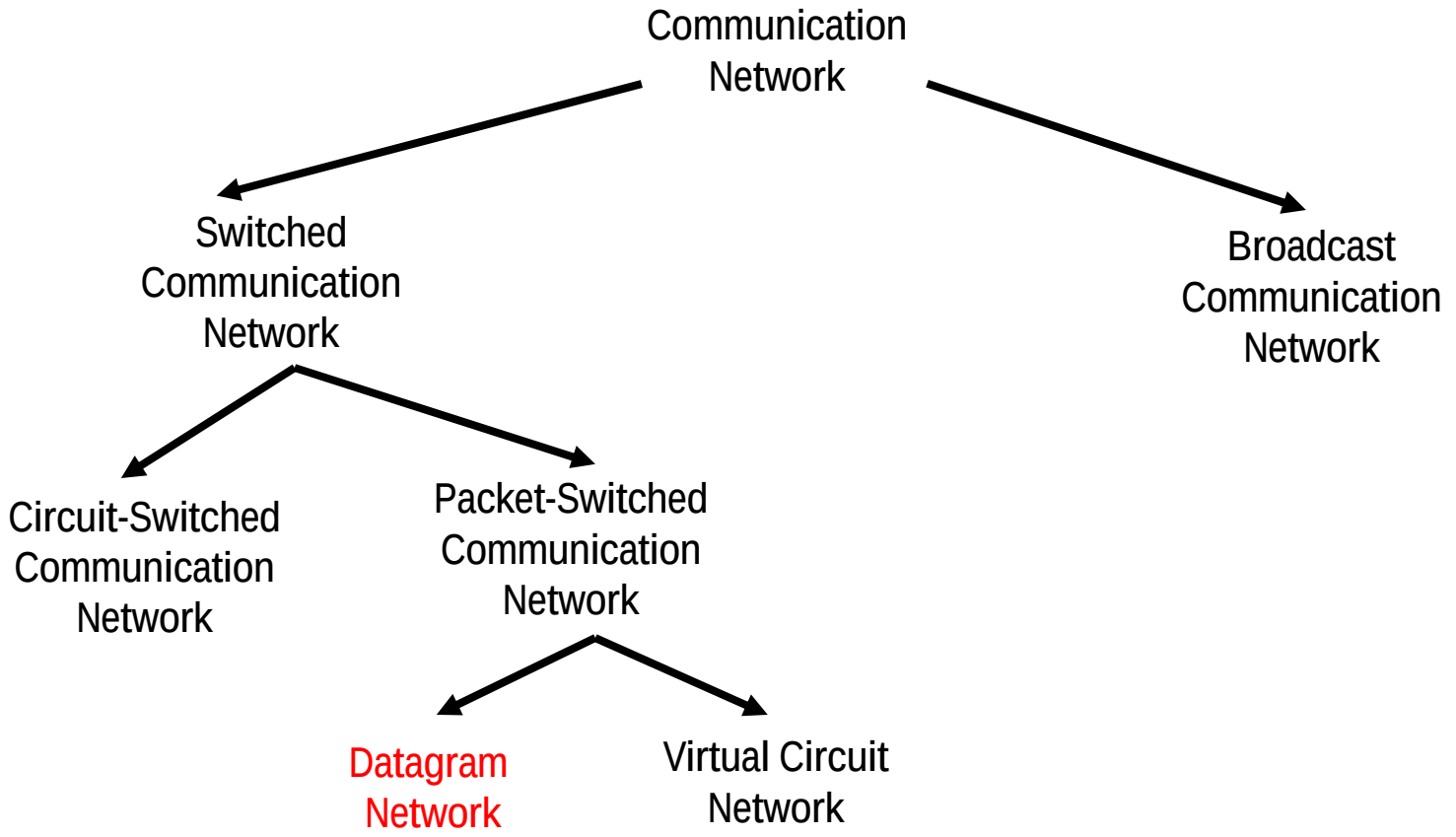
- With packet switching, packets are received, queued, and forwarded. Referred to as “Store and Forward”.



Datagram Network

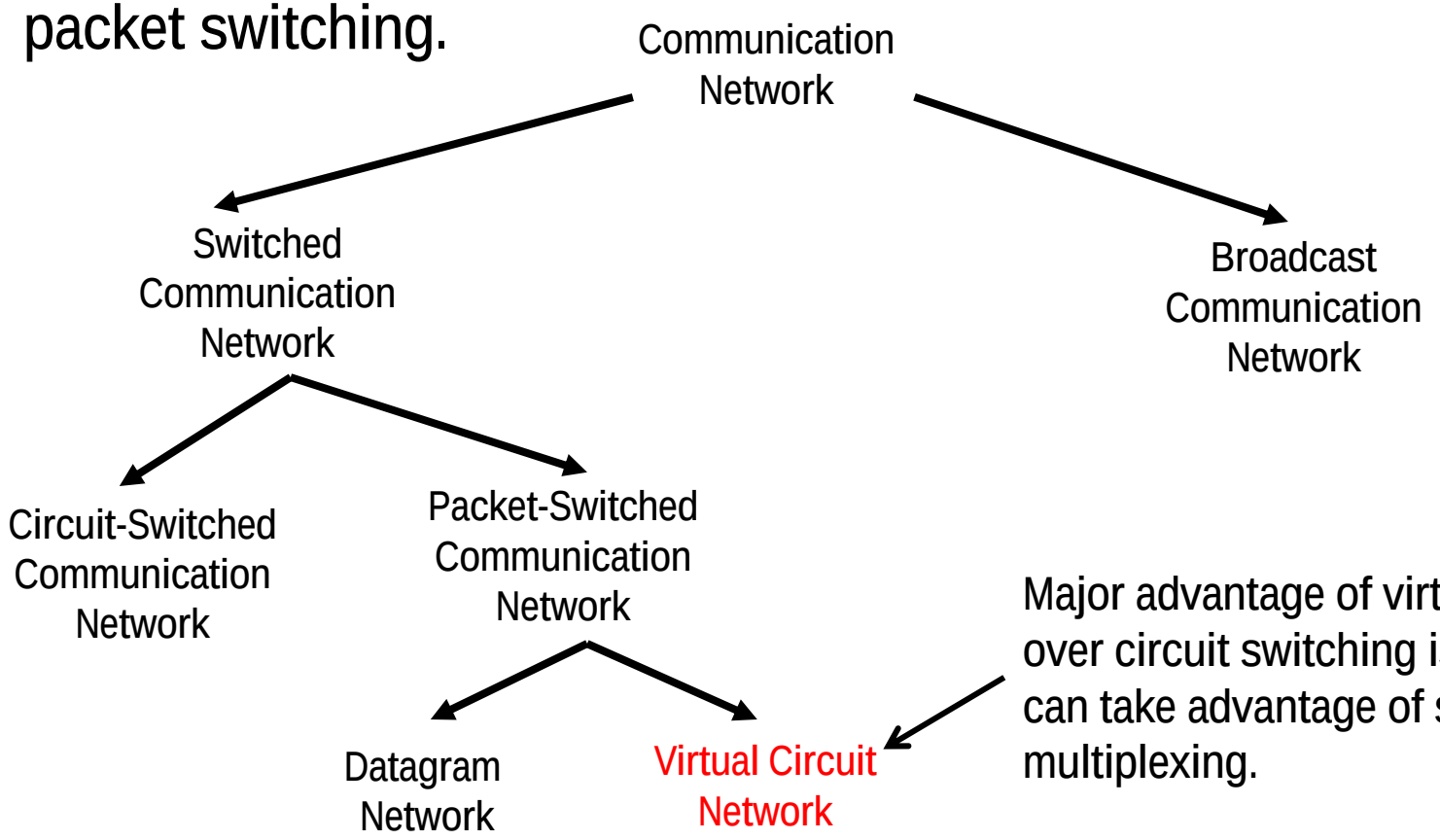


- In a datagram network (such as the Internet), each packet is independently forwarded. No pre-reservation of bandwidth or route.



Virtual Circuit Network

- In a virtual circuit network, a path is pre-chosen and all packets follow that path and arrive in order. May include reservation of resources. A hybrid of circuit switching and packet switching.



Major advantage of virtual circuit over circuit switching is that VC can take advantage of statistical multiplexing.

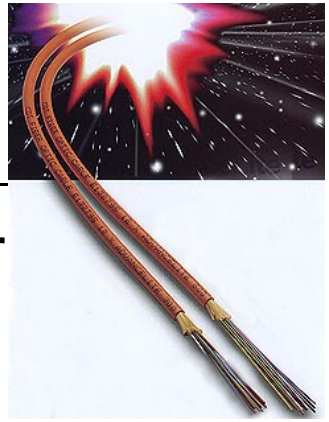
Frame Relay

- The first network we'll look at is Frame Relay, a virtual circuit network.
- In the late 60s and early 70s it became obvious that a system for transporting data was needed. Time sharing computers were becoming common and users were accessing them from async TTY type terminals.
- Since the traffic was async, the network had to guarantee delivery – there's no error checking or re-transmission in async. The network had to operate up to Layer 4.
- At that time, communication links were slow. Many were analog modem links and a 56Kbps line was considered high speed.

X.25

- The ITU responded by defining the X.25 network system.
- X.25 implemented all the way to Layer 4 in each node. It acknowledged packets across each link and did not forward packets until they were received correctly.
  - Added a lot of complexity and processing to the network.
  - End-to-end transmission was slow because of this processing.
- A number of networks were implemented using X.25:
  - Telnet, Tymnet, CompuServe, Euronet, and several others
- While X.25 was good for async traffic, users who had higher level protocols, such as IBM's SDLC, wanted a faster network.
- The answer was Frame Relay.

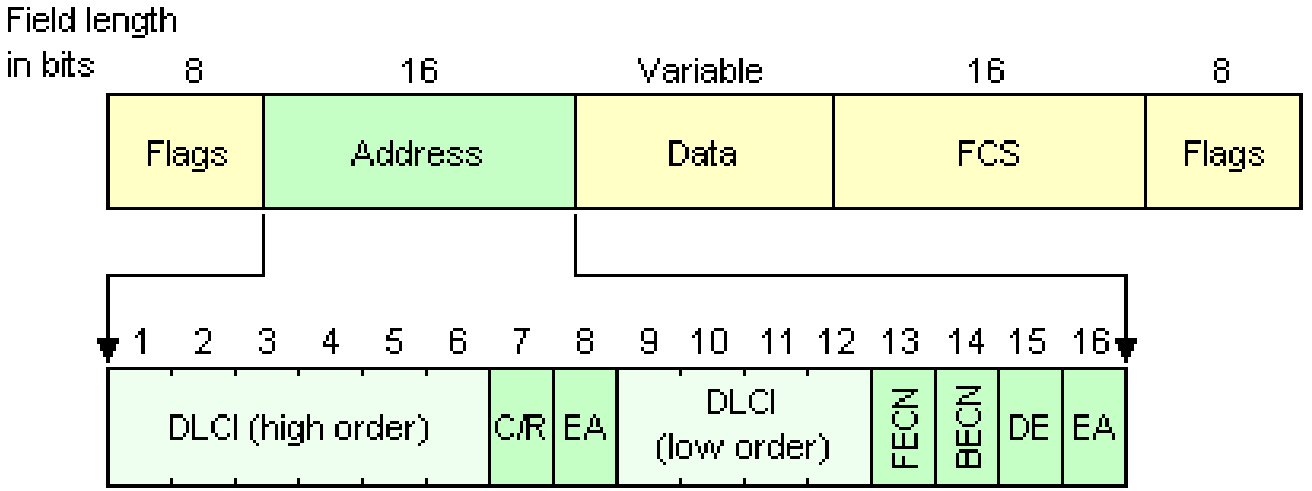
Frame Relay



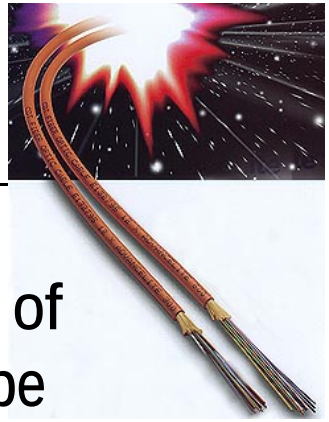
- Frame Relay was specifically designed to operate over digital links which had good BER.
- It was designed as a replacement for leased lines.
  - Permanent virtual circuits (PVC).
  - But provided for sharing of bandwidth because multiple users could be allocated PVCs over the same physical line.
- A layer 2 protocol – no error recovery. End points implemented a layer 4 protocol to guarantee delivery.
- Some protocols carried:
  - IBM's SNA.
  - Ethernet, to link physically separated networks. Note that Ethernet requires a layer 3 protocol in the host.
  - IP, with TCP in the hosts.

Frame Relay Frame

- Basically, an HDLC frame. The flag is 0111 1110 (0x7E).
- Zero bit insertion is used to prevent strings of 1s longer than 5
- The FCS is a 16 bit CRC.
- A frame may not exceed 8192 octets between flags, but the standards recommend 1600 octets.

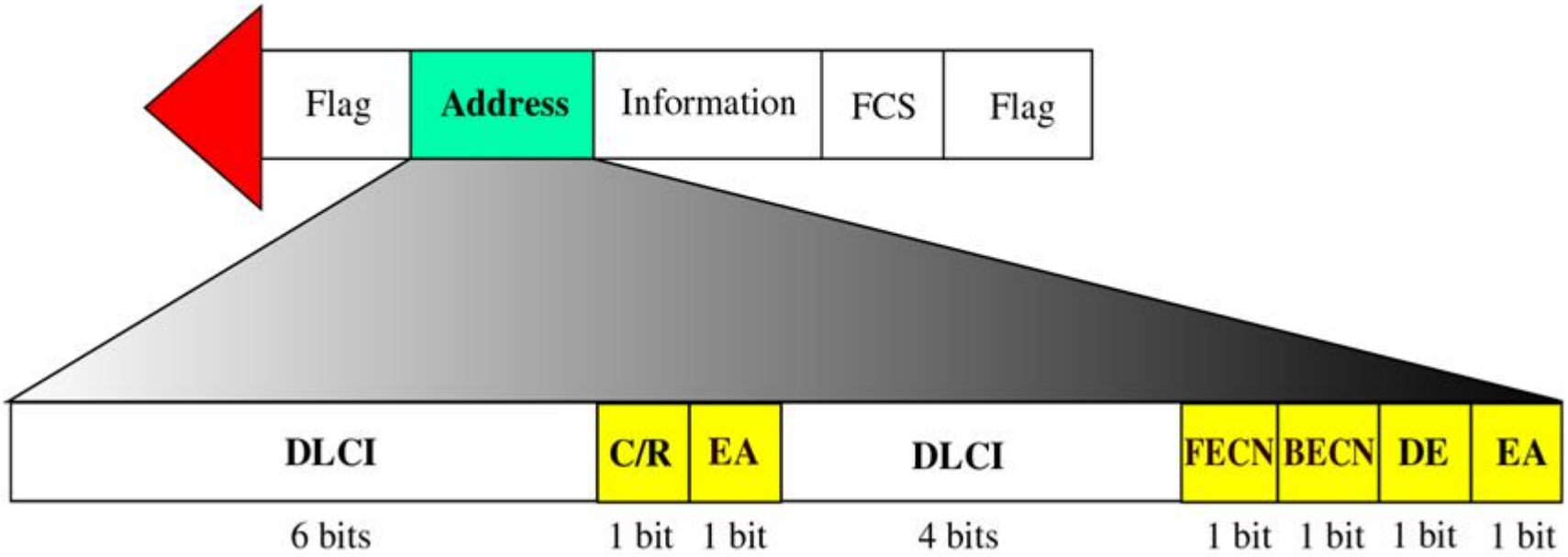


Frame Relay Frame



- Looking into the 16 bit address field, we find a number of fields as shown below. The meaning of the fields will be explored in the next few slides.

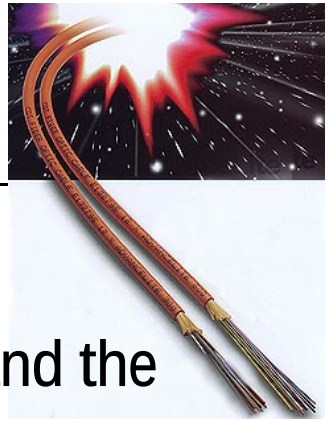
C/R: Command/response
EA: Extended address
FECN: Forward explicit congestion notification
BECN: Backward explicit congestion notification
DE: Discard eligibility
DLCI: Data link connection identifier



Frame Relay Address Fields

- The Data Link Connector Identifier (DLCI) identifies the virtual connection. While it's shown as a 10 bit field, larger fields are possible, called extended DLCI fields.
- For the 10 bit field, 1024 addresses are possible (0 to 1023).
  - Addresses 0 to 15 and 1008 to 1023 are reserved, leaving 16 to 1007 (992 addresses) for service providers to assign.
- We'll examine how the DLCI is used to route traffic later.
- The Command/Response (C/R) bit is not currently defined.
- The Extended Address (EA) bits indicate if a longer DLCI is being used. Current implementations only use the 10 bit DLCI.

Frame Relay Address Fields



- The Forward Explicit Congestion Notification (FECN) and the Backward Explicit Congestion Notification (BECN) bits are used to notify users, routers and/or frame relay switches that congestion is occurring at a node.
  - Allows devices upstream to slow down the sending of traffic.
  - Allows devices downstream to take action which might slow down traffic. For example, higher level protocols might slow down acknowledging packets, which would slow down the sending. No modern protocols take advantage of the FECN.

Routing in a Frame Relay Network

- A DLCI defines a virtual circuit from the user to the network.
- Multiple virtual circuits can share the same physical line.
  - So if a company wants to have connections to three different sites, only one physical line, such as a T1 line, is needed. Three different DLCIs will be used.
- DLCIs only have local significance – from the user to the network. Once the frame gets to the first Frame Relay switch, the DLCI will be changed for the next leg of the circuit.
- When a customer contracts with a provider for a Frame Relay circuit, the provider defines the path that the circuit will take through the network and causes routing tables to be built in the appropriate switches.

Routing in a Frame Relay Network

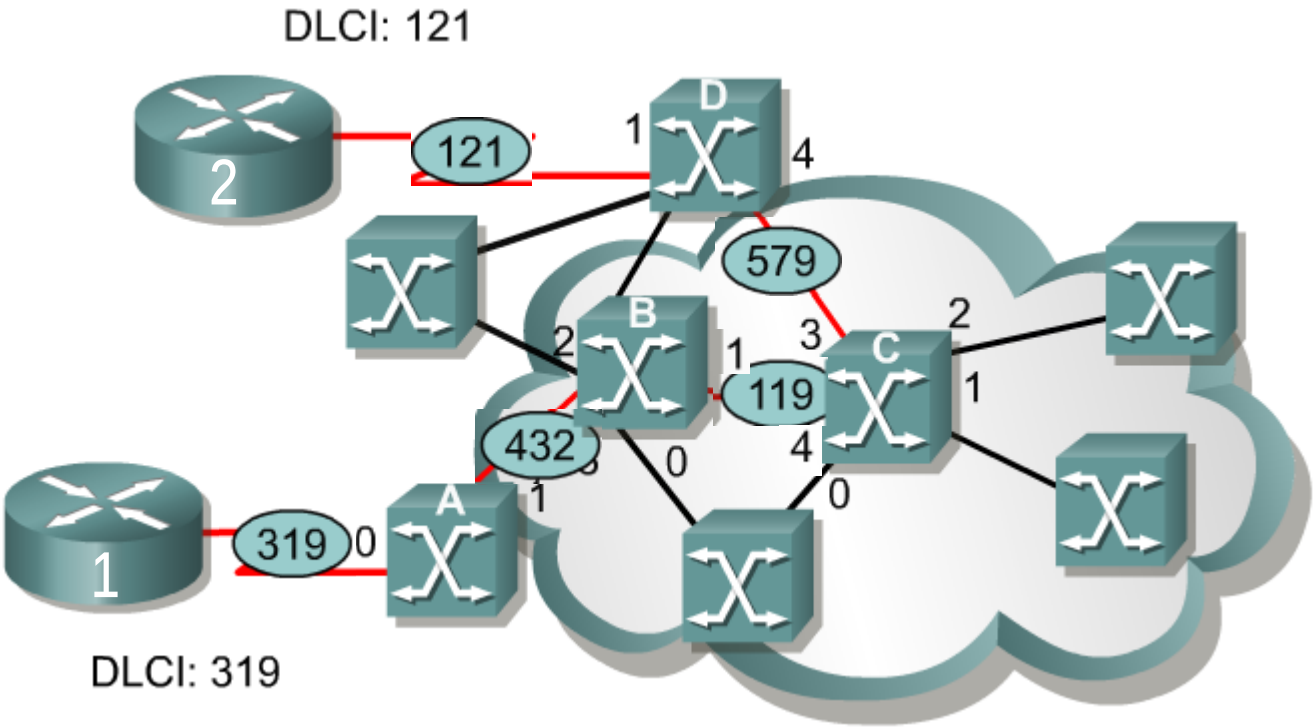
- Here are example routing tables for routing DLCI 319 from User 1 to User 2.
- User 1 sends a frame to switch A with DLCI = 319
- Router A looks in it's table and sees that the frame should be sent out on port 1 with DLCI 432.

A			
VC	Port	VC	Port
319	0	432	1

B			
VC	Port	VC	Port
432	3	119	1

C			
VC	Port	VC	Port
119	4	579	3

D			
VC	Port	VC	Port
579	0	121	1



Routing in a Frame Relay Network

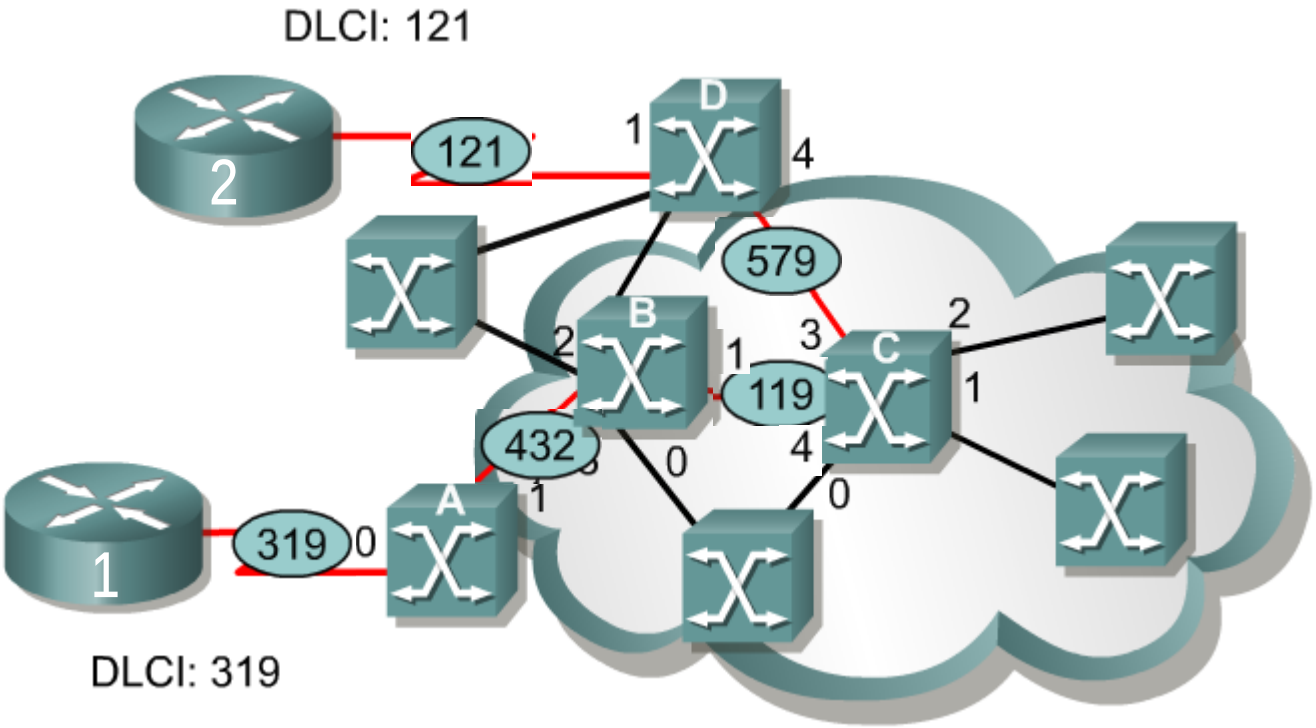
- Switch B then receives the frame with DLCI 432 and sends it out on port 1 with DLCI = 119
- Switch C receives the frame, does a lookup and sends it out on port 3 with DLCI = 579.

A			
VC	Port	VC	Port
319	0	432	1

B			
VC	Port	VC	Port
432	3	119	1

C			
VC	Port	VC	Port
119	4	579	3

D			
VC	Port	VC	Port
579	0	121	1



Routing in a Frame Relay Network

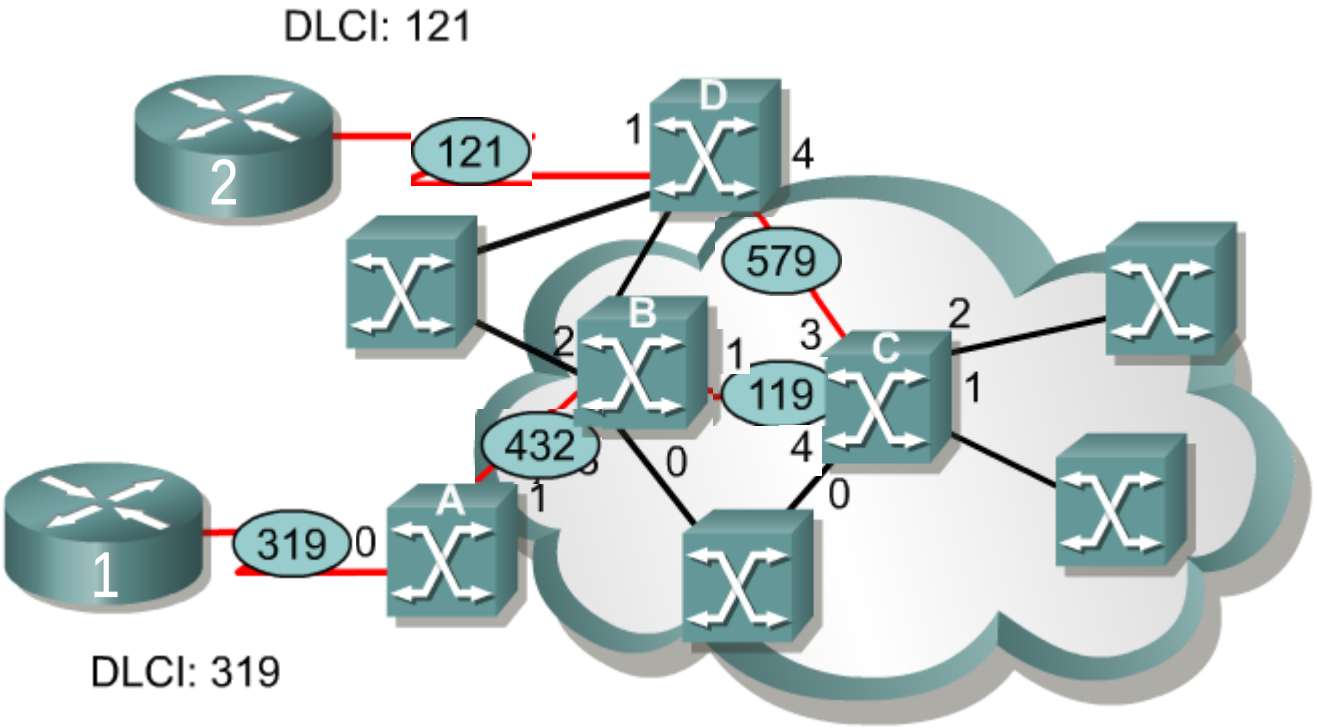
- Switch D receives the frame, does a lookup, and sends the frame out on port 1 with DLCI = 121.
- User 2 knows that frames received with DLCI = 121 are for the circuit between User 1 and itself.

A			
VC	Port	VC	Port
319	0	432	1

B			
VC	Port	VC	Port
432	3	119	1

C			
VC	Port	VC	Port
119	4	579	3

D			
VC	Port	VC	Port
579	0	121	1



Routing in a Frame Relay Network

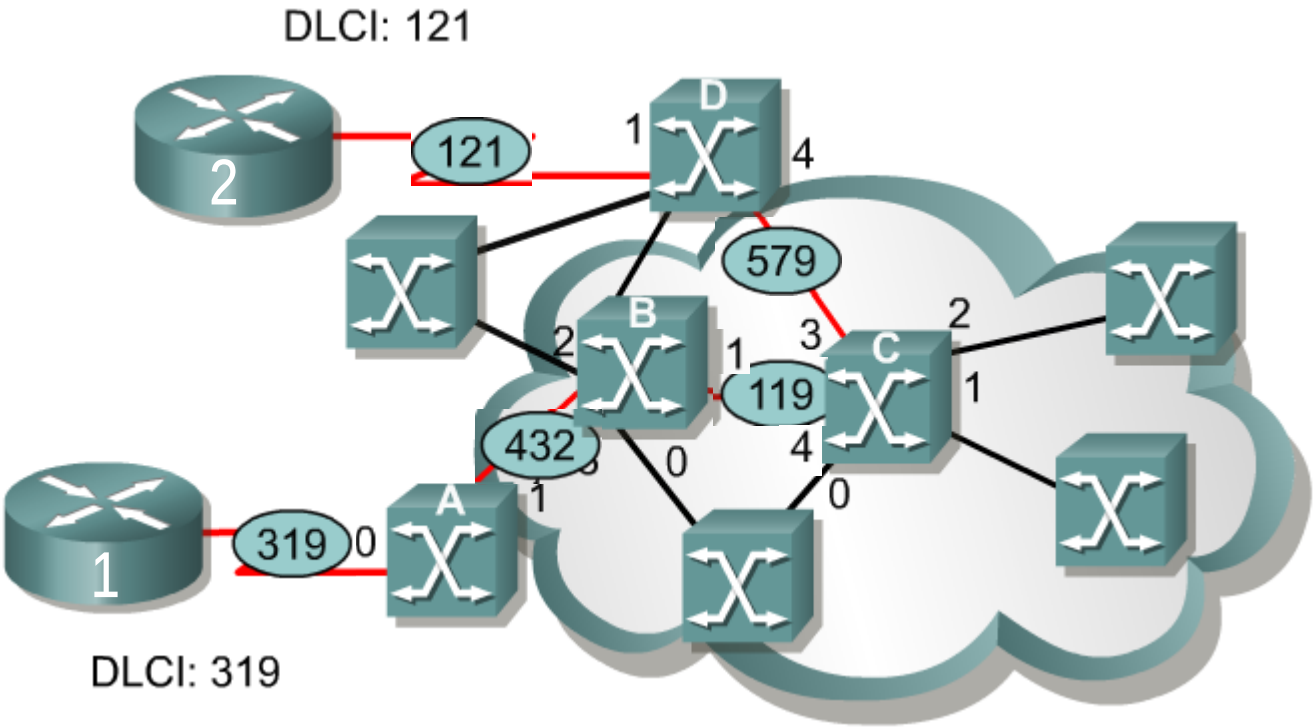
- Note that it might be quicker to send the frames from Switch B to Switch D but the route is what the provider defined and all frames between User 1 and User 2 will take the same route.

A			
VC	Port	VC	Port
319	0	432	1

B			
VC	Port	VC	Port
432	3	119	1

C			
VC	Port	VC	Port
119	4	579	3

D			
VC	Port	VC	Port
579	0	121	1



Problems with Frame Relay

- Frame relay was successful for what it was designed for – to be a replacement for leased lines. But it has certain limitations:
  - Since it supports fairly long frames, it does not support "time sensitive" traffic, such as voice, very well. A queued up long frame may delay a voice frame, which will cause jitter in the voice traffic.
  - While switched virtual circuits were defined for frame relay, they were mostly never implemented. Rather than modify frame relay networks, the decision was made to develop a new system – Asynchronous Transfer Mode (ATM).
  - Providers wanted a network that could carry all of their traffic: voice, data and video. Frame relay could not do that. Enter ATM.

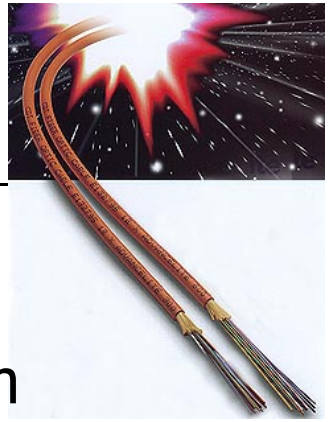
Asynchronous Transfer Mode (ATM)

- ATM was developed over a number of years but the final ITU "standard" was adopted in 1995 (I.361, revised in 1999).
- The goal was to develop a network that would integrate voice and data communications. Remember, back then, voice was still the majority of the traffic in the network.
- Probably because of the predominance of voice traffic, the designers came up with a system that had many similarities to the existing voice network:
  - The system is "circuit" based. Traffic flows through permanent virtual circuits (PVC) or switched virtual circuits (SVC).
  - The system was optimized for carriage of voice: to provide low latency and low jitter to voice traffic. That's the primary reason for the small cell size.

Synchronous and Asynchronous Muxing

- In TDM circuits, such as a T1 circuit, multiple channels are multiplex on the circuit.
- Each channel is allocated a dedicated time slot and that time slot is used “synchronously”. The receiver can extract a channel by knowing the time slot.
- In ATM, each channel is not allocated a dedicated time slot. The traffic is put into slots as needed. The slots are used “asynchronously”.
- Since the data cannot be identified by a specific slot, it must carry information in a header to identify it so that it can be routed to the receiver and so the receiver can extract each channel.
- ATM is a form of statistical multiplexing.

The ATM Cell

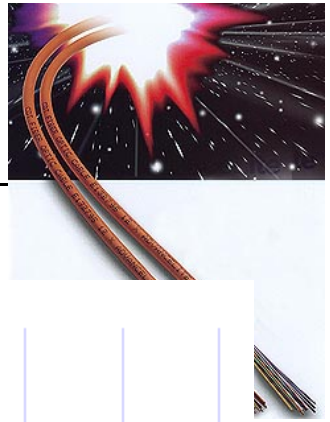


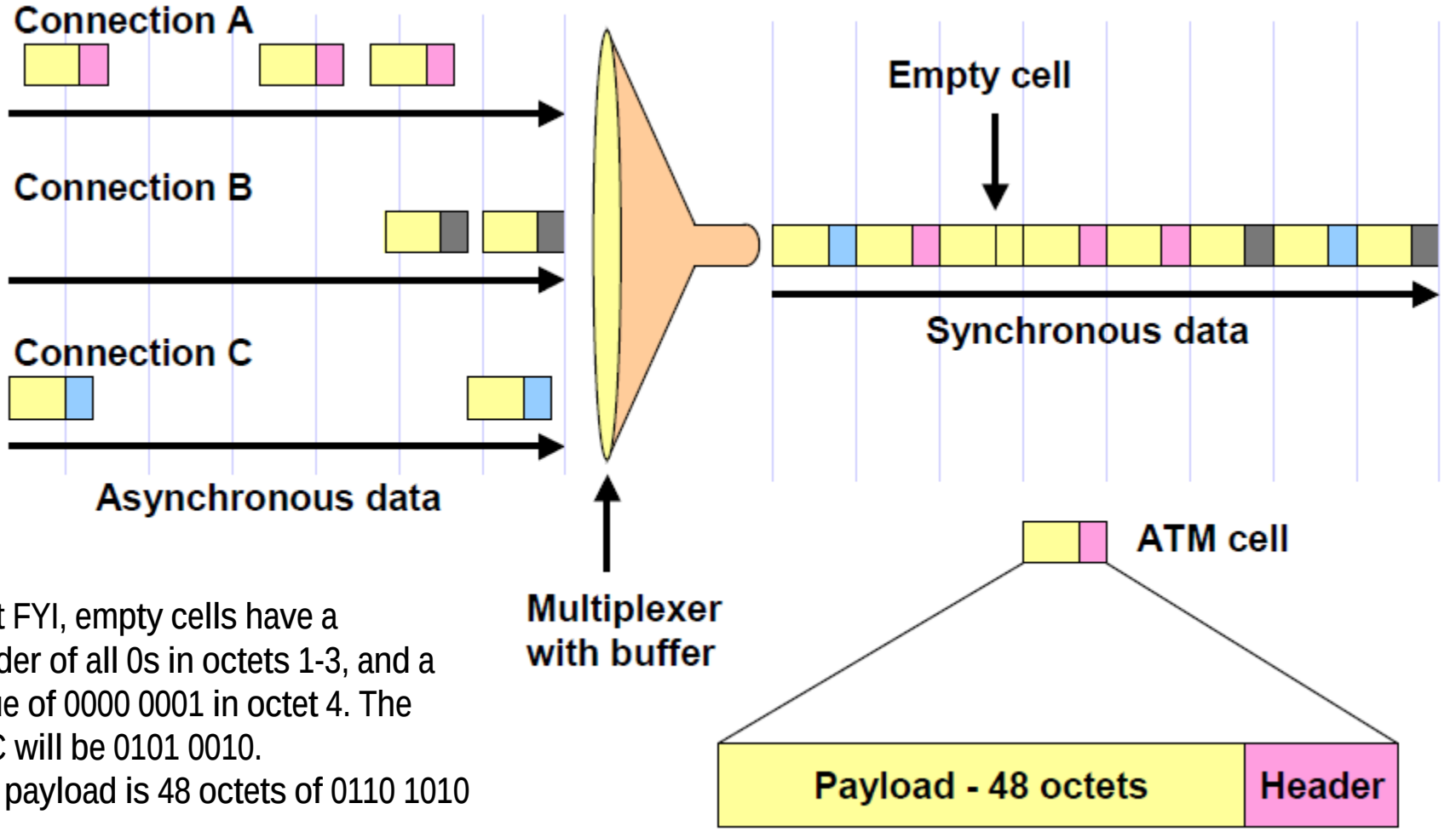
- At the time ATM was defined, a high link speed was 155mbps over fiber (OC-3). And many links were much slower: DS3 (45Mbps), and DS1 (1.544Mbps).
- Any large frame (as in frame relay) would tie up the line for a significant time while it was being transmitted.
- The designers of ATM, therefore, decided to use a relatively small fixed length cell:
  - The US wanted the data portion to be 64 octets.
  - Europe, especially France, wanted 32 octets.
  - They compromised on 48 octets.
  - Adding a 5 octet header gives us a 53 octet cell.

ATM Cells

- Why a fixed size cell?
  - Primarily to control jitter in voice traffic. If variable sized packets were used, some voice packets would be delayed while large data packets were transmitted.
  - Would require buffering at the receiver to accommodate the jitter and that would increase delay. Delay then requires echo cancellers on the line and increases “talkover”, making conversation awkward.
  - Also makes it easier, and faster, to switch the traffic.
- Why so small a cell?
  - Voice cells get priority, but have to wait for the cell being processes to get onto the line. Less waiting if the cell is small, means less jitter.
  - Negative is high amount of overhead (header size to payload size).

ATM Multiplexing





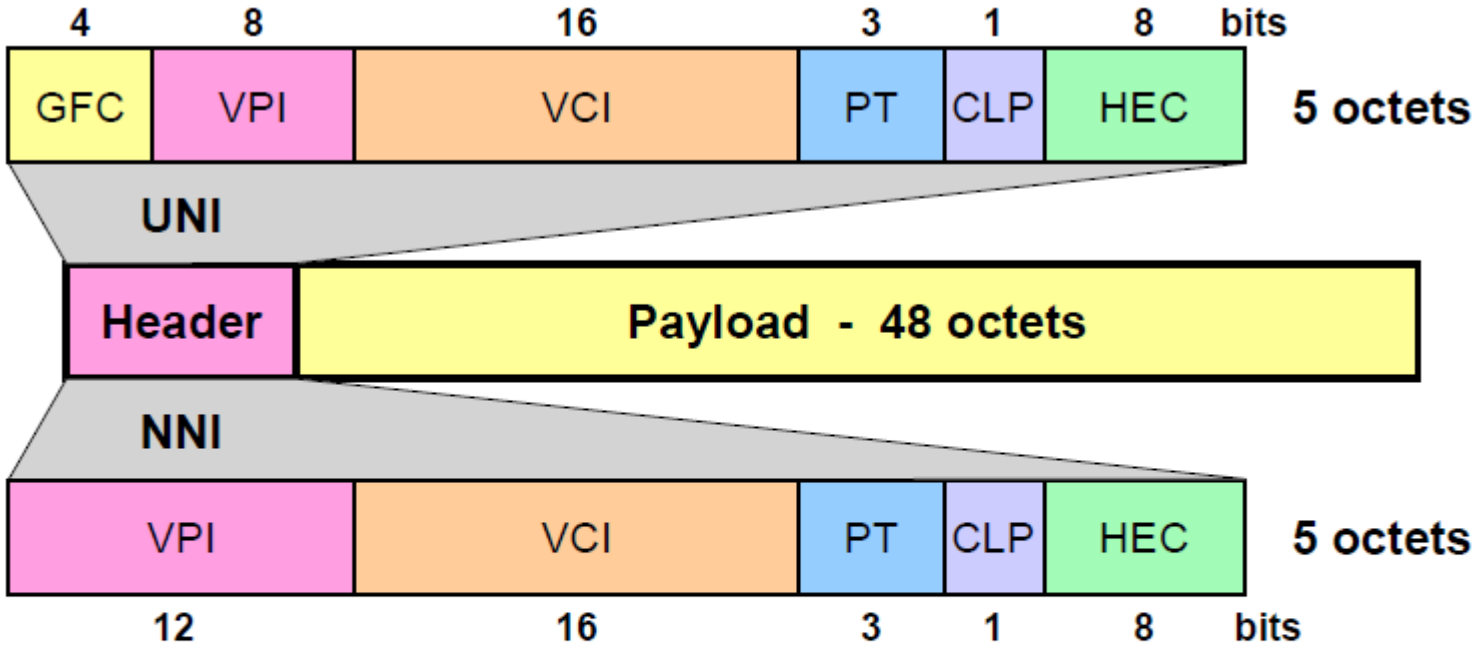
Just FYI, empty cells have a header of all 0s in octets 1-3, and a value of 0000 0001 in octet 4. The HEC will be 0101 0010.
The payload is 48 octets of 0110 1010

ATM

- When transmitting cells, the ATM node will take cells, either on a first-in, first-out, or priority basis and put them into slots on the line.
- If there are not enough data cells, empty cells are sent to maintain synchronous communications.
- If there are too many cells to output to the line, some will be dropped, according to the service level agreement with the provider. This is called “policing” or usage parameter control (UPC).

ATM Cell Header

- Two formats, one for User-Network Interface (UNI) and one for Network-Network Interface (NNI).



UNI	User-Network Interface	NNI	Network-Network Interface
GFC	Generic Flow Control	PT	Payload Type
VPI	Virtual Path Identifier	CLP	Cell Loss Priority
VCI	Virtual Channel Identifier	HEC	Header Error Control

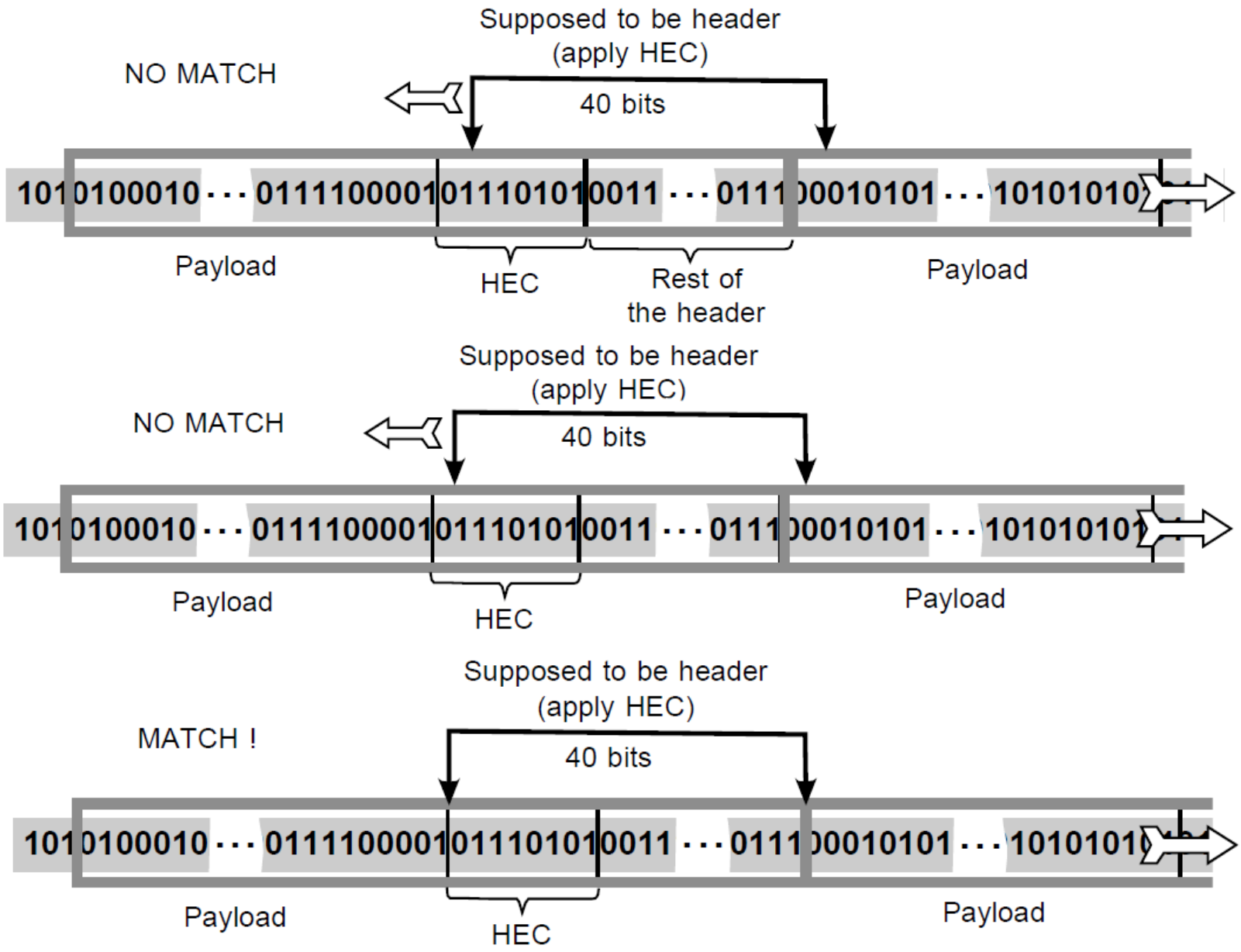
ATM Cell Header

- GFC – Generic Flow Control (UNI only). This field is not really used. It's never been defined.
- VPI – Virtual Path Identifier. Identifies a virtual path between two ATM nodes.
- VCI – Virtual Channel Identifier. Identifies a virtual channel between ATM nodes or within a VPI.
- PT – Payload Type. Indicates operation and maintenance or resource management, or cells with or without congestion.
- CLP – Cell Loss Priority. 0 = higher priority, 1 = lower priority.
- HEC – Header Error Control. An 8 bit CRC over the header. Polynomial $x^8 + x^2 + x + 1$. This will detect most (84%) multiple bit errors and using the BCH algorithm, will correct single bit errors.

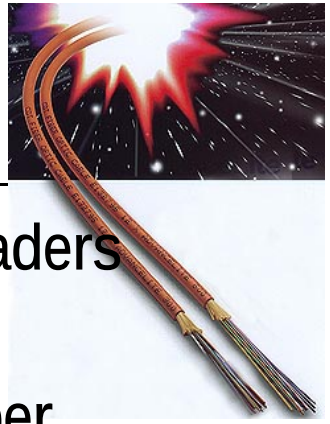
Finding the Start of an ATM Cell

- Note that there's no framing character in the ATM cell header. So how do we find the start of the cell.
- The receiver examines the incoming bit stream bit by bit and computes the CRC over 4 octets (32 bits). It then compares the computed CRC to the next octet.
- If the CRC checks, the receiver looks ahead 53 octets and does the same computation. If the CRC checks for three cells, the receiver assumes it has cell synchronization.
- If not, the receiver moves one bit and goes through the process again, until it finds three consecutive cells that check.
- Loss of cell sync is declared if three consecutive cells do not CRC check.

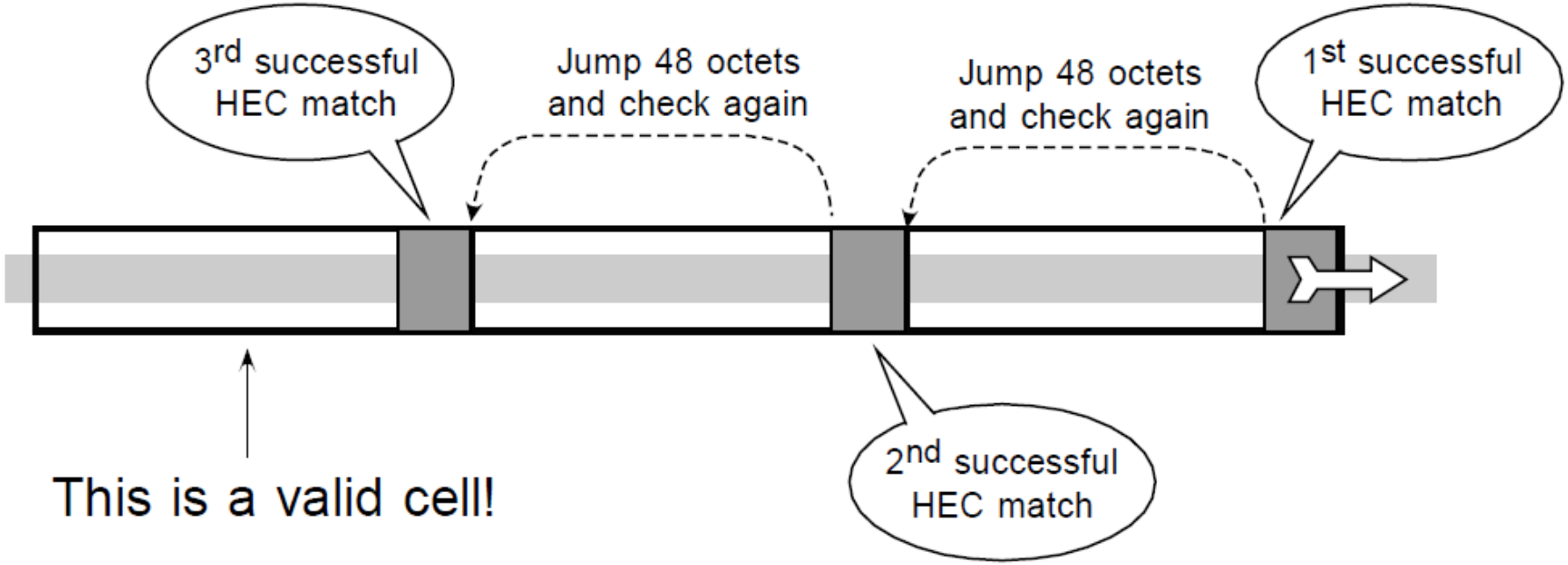
Finding the Start of an ATM Cell



Finding the Start of an ATM Cell

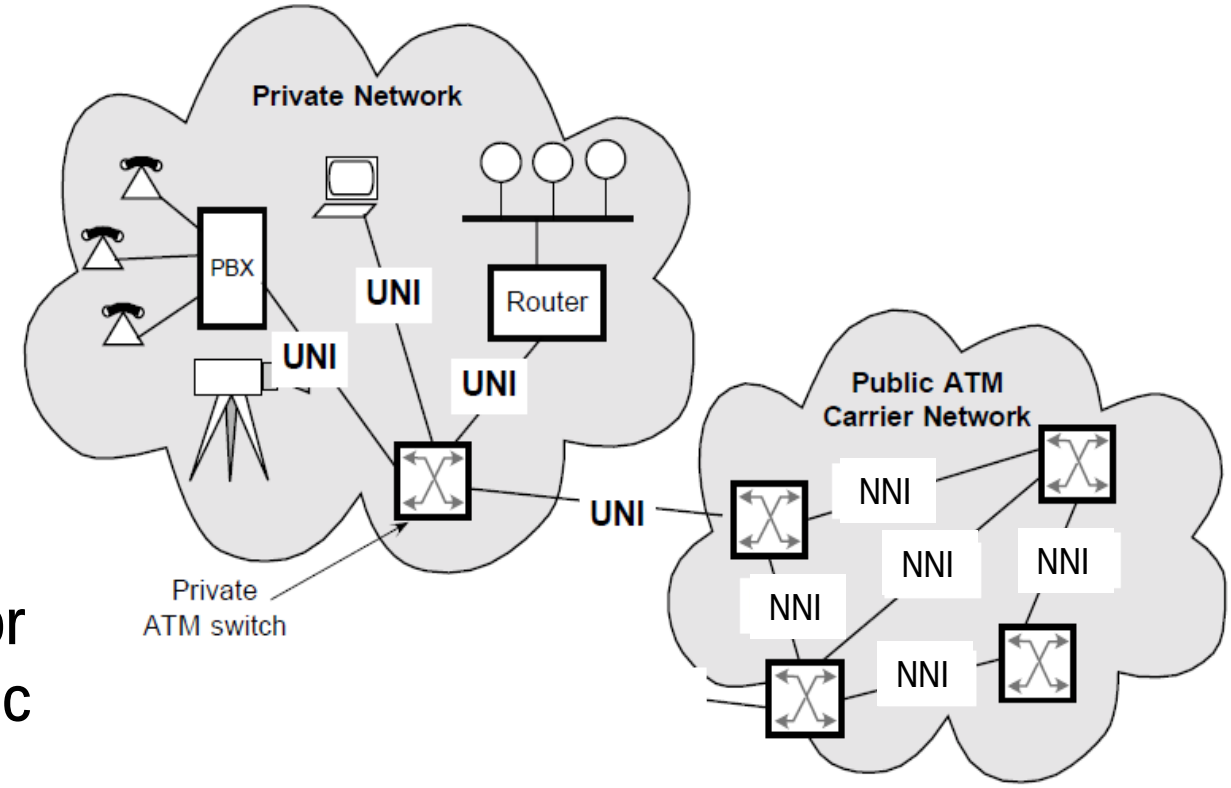


- To enter the synchronized state, three consecutive headers must CRC check.
- Not all networks use 3. Some could use a larger number.
- Same for declaring loss of sync. Some networks may use a number greater than 3 for non-CRC check.



UNI and NNI

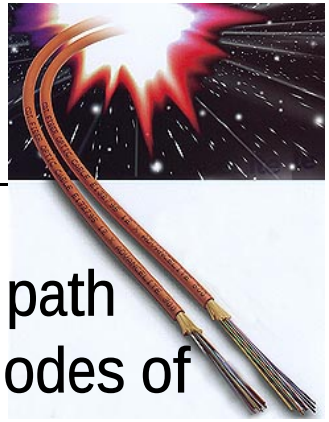
- Note that the cell header has two formats: One for the User-Network Interface (UNI) and one for the Network- Network Interface (NNI).
- The UNI is used by a subscriber on the connection to the network or from a private network to a public network.
- The NNI is used within a network or between two public networks.



Virtual Channels and Virtual Paths

- ATM split the “address” of the cell into two fields, the Virtual Path Identifier (VPI) and the Virtual Channel Identifier (VCI). Let’s discuss why they did that.
- A virtual channel connection (VCC) is a logical end-to-end connection from one end user to another.
- A virtual channel connection is made up of virtual channel links (VCL), which are the connections between the nodes of the ATM network.
- Each VCL is identified by a virtual channel identifier (VCI). There will be a different VCI for each of the VCLs (the VCI will change every time the cell goes through a virtual channel switch).

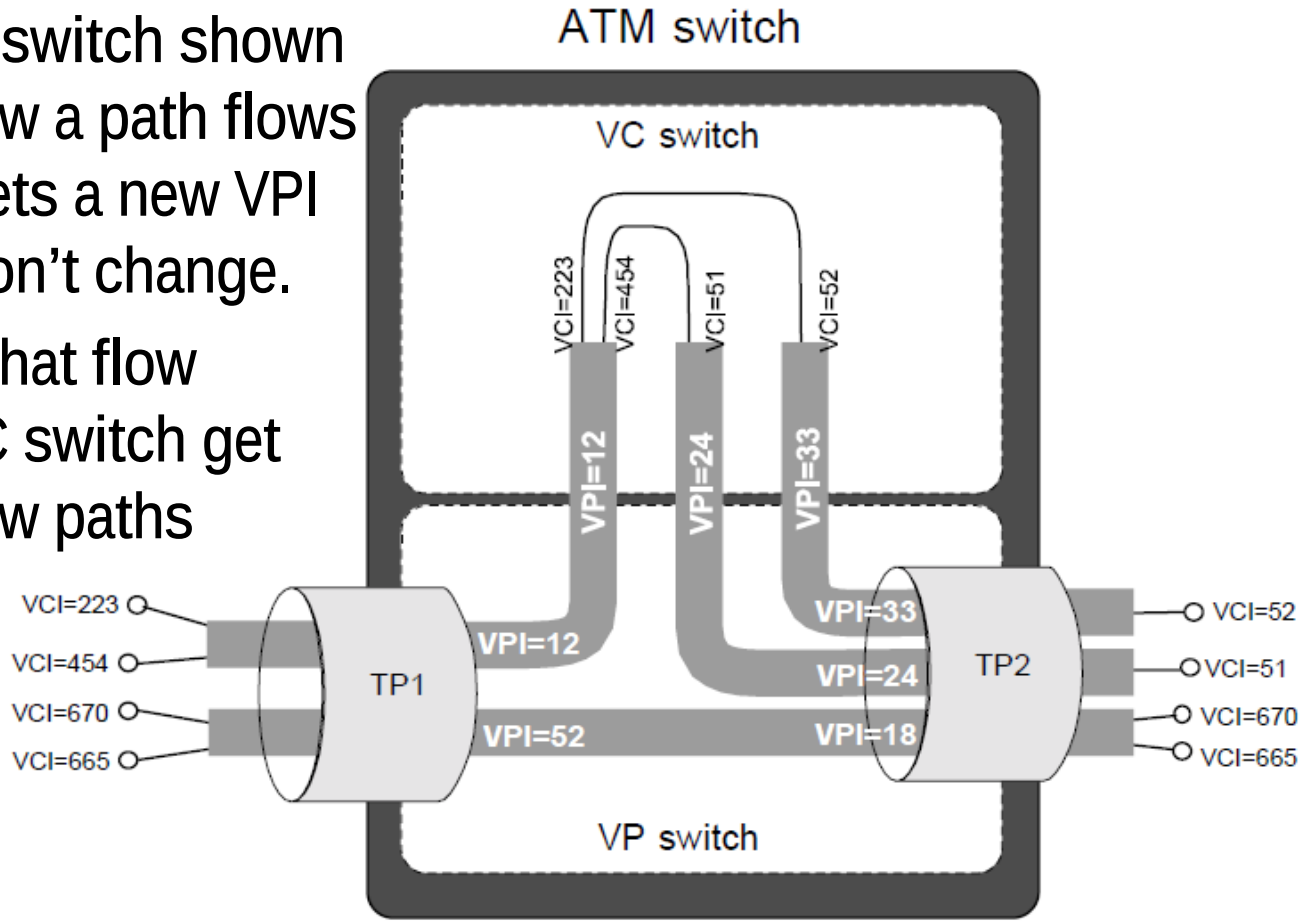
Virtual Channels and Virtual Paths



- A virtual path connection (VPC) is made up of virtual path links (VPL), which are the connections between the nodes of the ATM network.
- Each VPL is identified by a virtual path identifier (VPI). There will be a different VPI for each of the VPLs (the VPI will change every time the cell goes through a virtual path switch).
- Only 8 bits are allocated to the VPI in the UNI because any one user is unlikely to have more than 255 paths.
- 12 bits, giving 4095 paths, are used for the NNI.
- In both cases, 16 bits are allocated to the VCI, giving 65,535 possible channels.

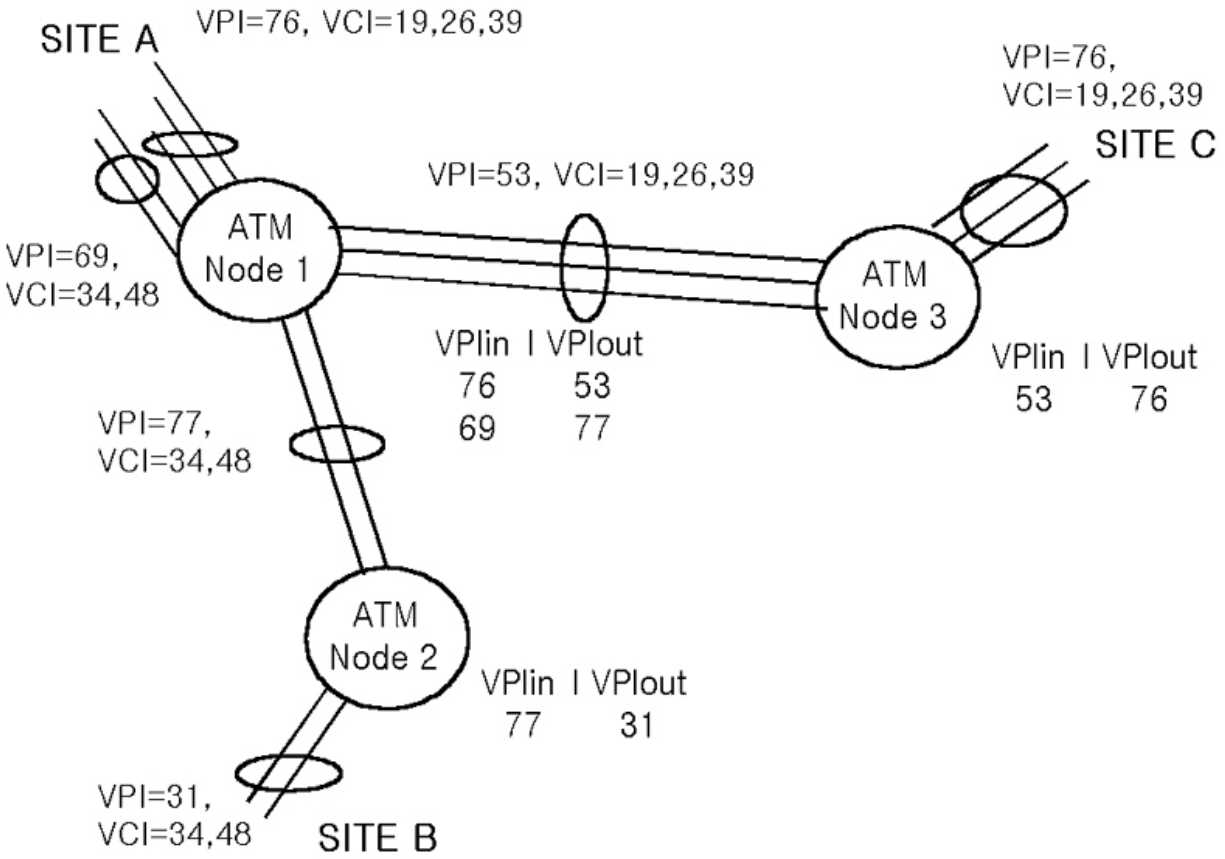
VC and VP Switches

- A virtual path switch only switches paths.
- A virtual channel switch only switches channels.
- The combined switch shown here shows how a path flows through and gets a new VPI but the VCIs don't change.
- The channels that flow through the VC switch get assigned to new paths and receive a new VCI.



VC and VP Switches

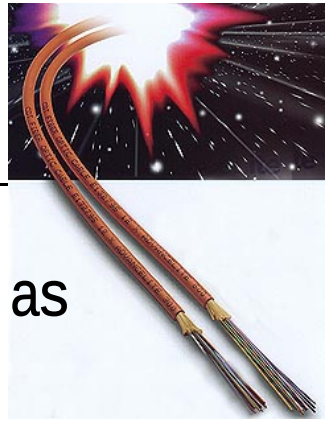
- Here's a look at virtual channels that can be switched through path switching only.
- From Site A to Site B, note that the VPI changes but the VCIs do not. Only the path has been switched.
- From Site A to Site C, again, the VPI changes at every switch, but the VCIs stay the same.
- This makes for fast switching.



Service Categories

- ATM offers five different service categories.
  - Constant Bit Rate (CBR) – priority 1. Gives a guaranteed amount of bandwidth to a virtual channel. Basically function like a T1 or DS3 circuit. Used for real time applications, such as voice channels.
  - Real-time Variable Bit Rate (rt-VBR) – priority 2. Provides support for real-time applications that are bursty in nature. An example might be a voice coder that implements silence suppression.
  - Non-real-time Variable Bit rate (nrt-VBR) – priority 3. Used for real time applications which are more tolerant of network delays.
  - Available Bit Rate (ABR) – priority 4. Intended for real-time applications which can tolerate some delay and cell loss.
  - Unspecified Bit Rate (UBR) – priority 5. A best efforts service. No guarantees at all.

Traffic Classes

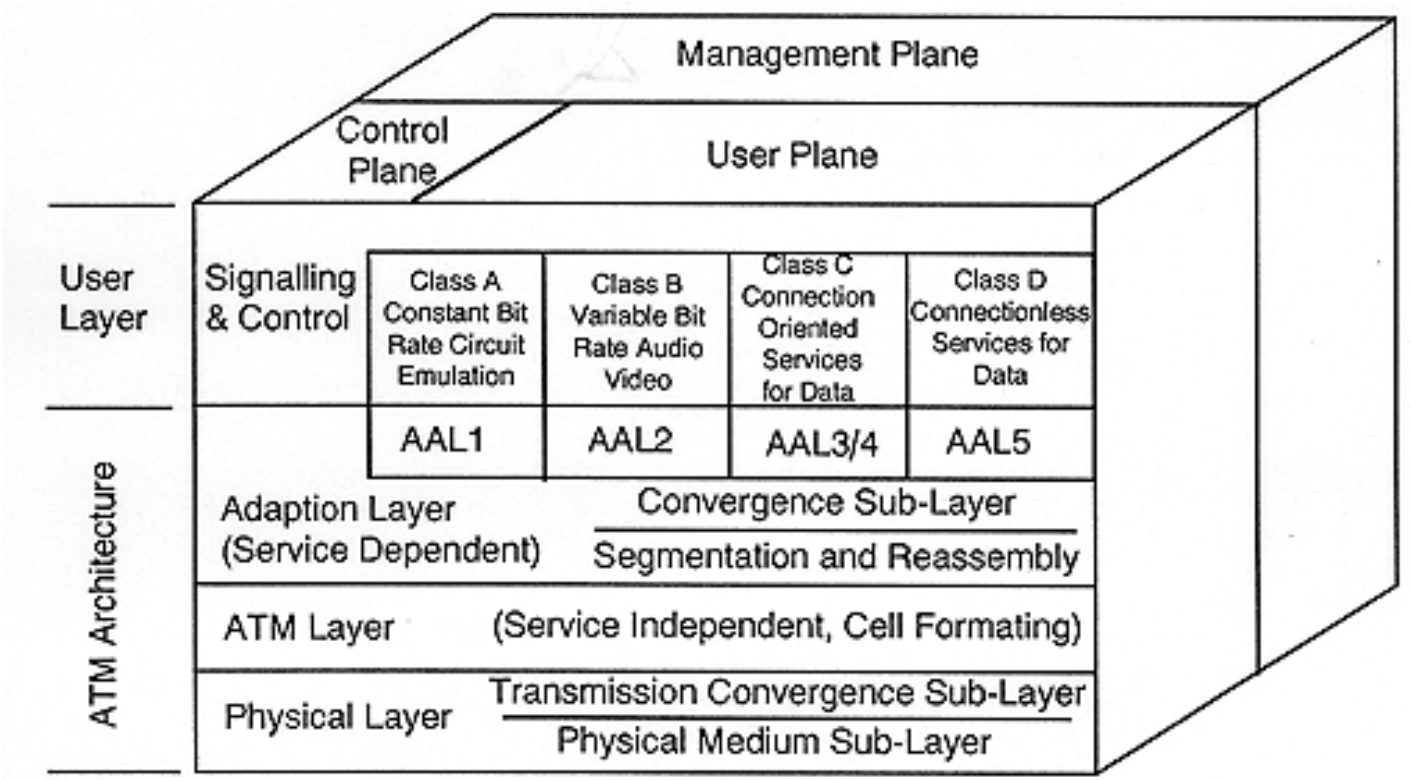


- ATM traffic is divided into four classes, A, B, C, and D, as shown in the figure.
- Characteristics of the classes are also shown.

Traffic Class	Class A	Class B	Class C	Class D
Timing relation between source and destination	Required		Not required	
Bit Rate	CBR	VBR		
Connection Mode	Connection-Oriented			Connectionless
AAL Type	AAL1	AAL2	AAL3/4 or AAL5	
Example Application	T-1, E-1 circuit emlation	Packet video, audio	FR, X.25	IP, SMDS

ATM Protocol Stack

- Given the four traffic classes, we can now see the ATM protocol stack, and we will begin to examine the ATM Adaption Layers (AAL)

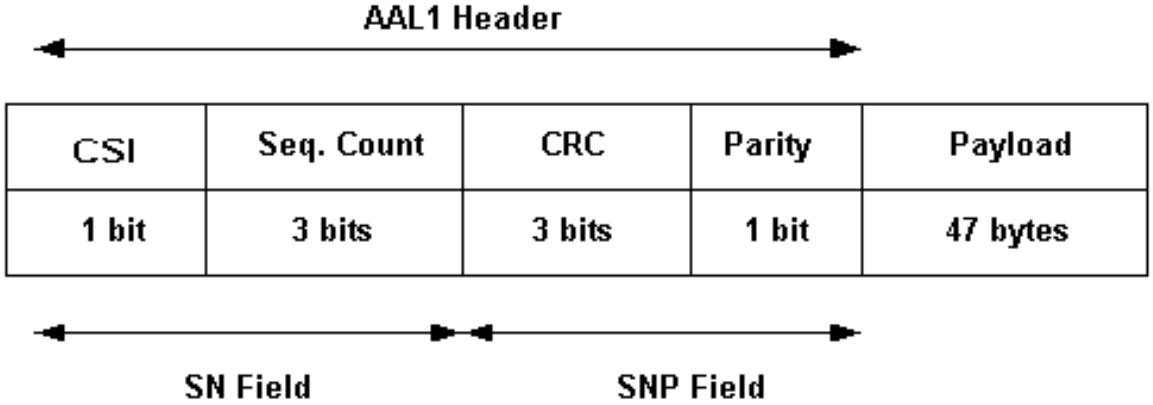


ATM Adaption Layer

- The ATM Adaption Layer consists of the Segmentation and Reassembly (SAR) layer and the Convergence Sublayer (CS).
- The function of the SAR is to fragment and reassemble variable length packets into 48 octet fixed length cells. On certain AAL types it will add a one octet header, leaving 47 octets from the higher layer. The five octet ATM headers are added at the ATM Layer.
- The Convergence Sublayer is responsible for adding headers (and trailers in certain circumstances) that will allow it to reassemble the original bit stream from the PDUs passed to it from the SAR. This varies depending on the type of AAL.

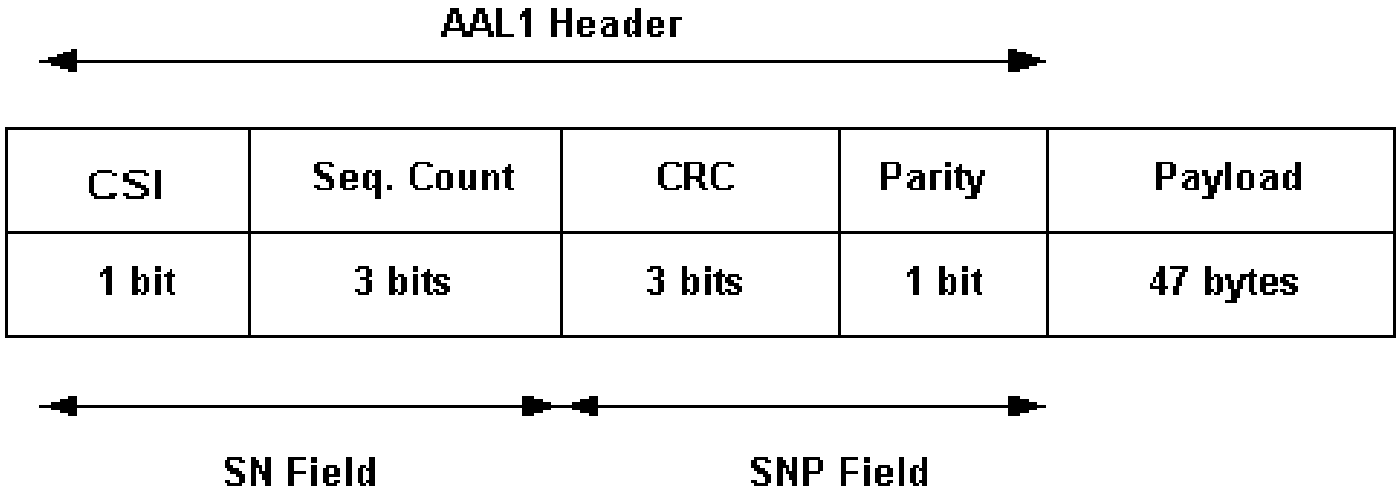
ATM Adaption Layers (AAL)

- The ATM Adaption Layer used for traffic does not define the service category but certain AALs are generally used for certain service categories.
- There are five AALs but only three see any real use – AAL1, AAL2, and AAL5.
- AAL1 was designed for CBR traffic, especially for circuit emulation to carry DS1 or E1 traffic.
- One octet is taken from the 48 octet Protocol Data Unit (PDU) for the AAL1 header.



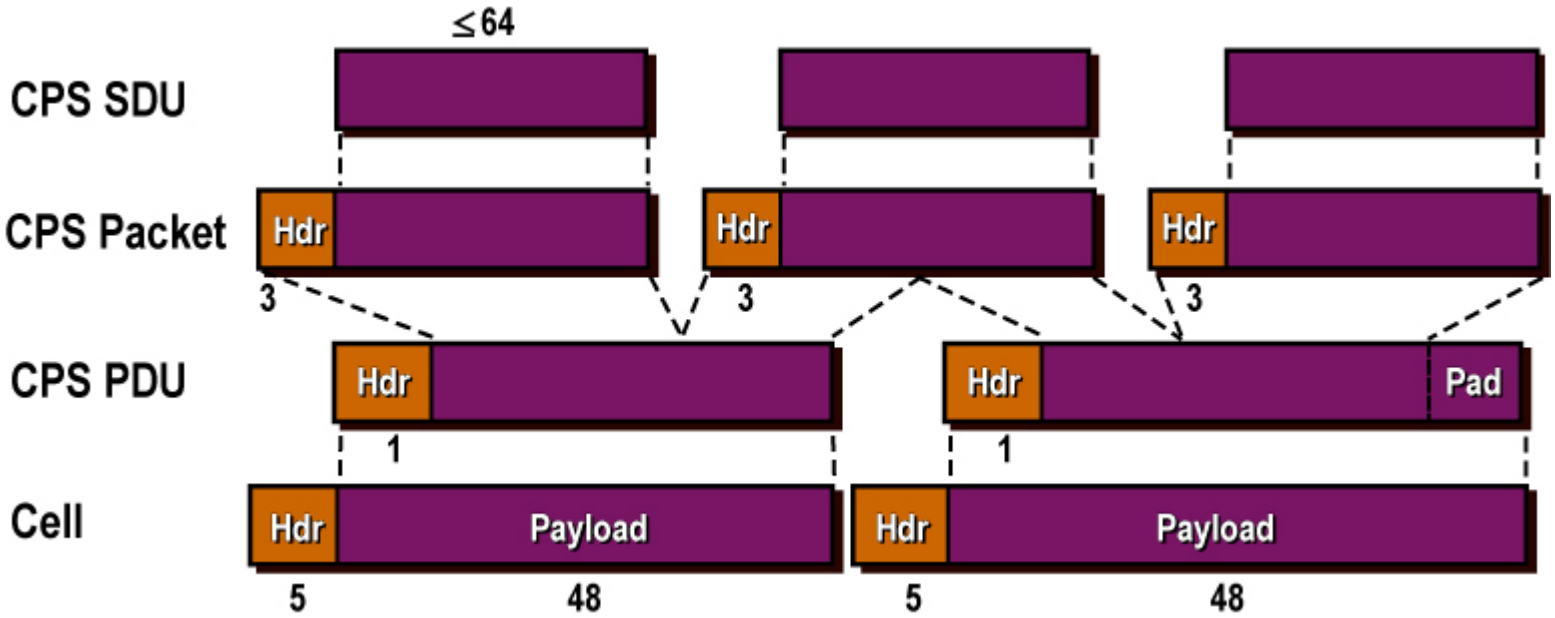
AAL1

- The first bit is the Convergence Sublayer Indicator (CSI). Basically, the CSI provides for clock synchronization.
- The Sequence Count (Seq. Count) is a modulo 8 counter to detect missing or out of order cells.
- The CRC protects the CSI and Sequence Count fields.
- The parity bit is even parity over the previous 7 bits.



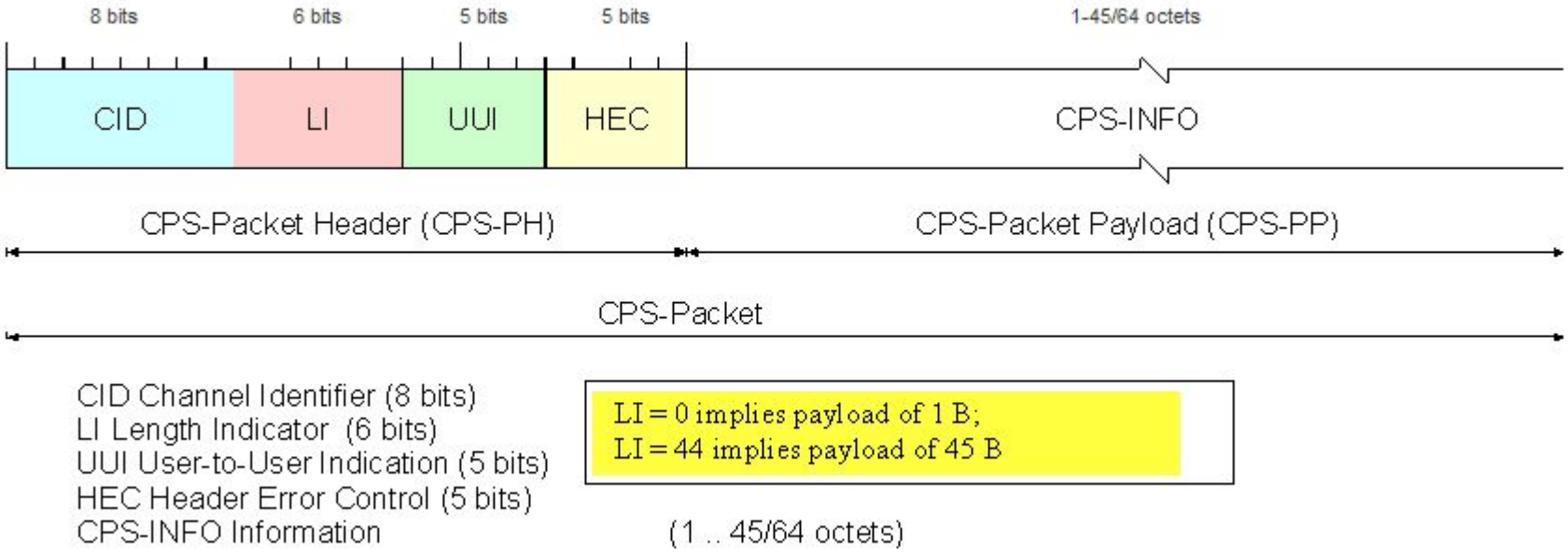
AAL2

- AAL2 is intended for variable bit rate (VBR) applications.
- First, the CS breaks the VBR bit stream into 45 to 64 octet Sublayer Data Units (SDU) and adds a 3 octet header. 45 octet segments are used the most. This function is called the Common Part Sublayer (CPS).



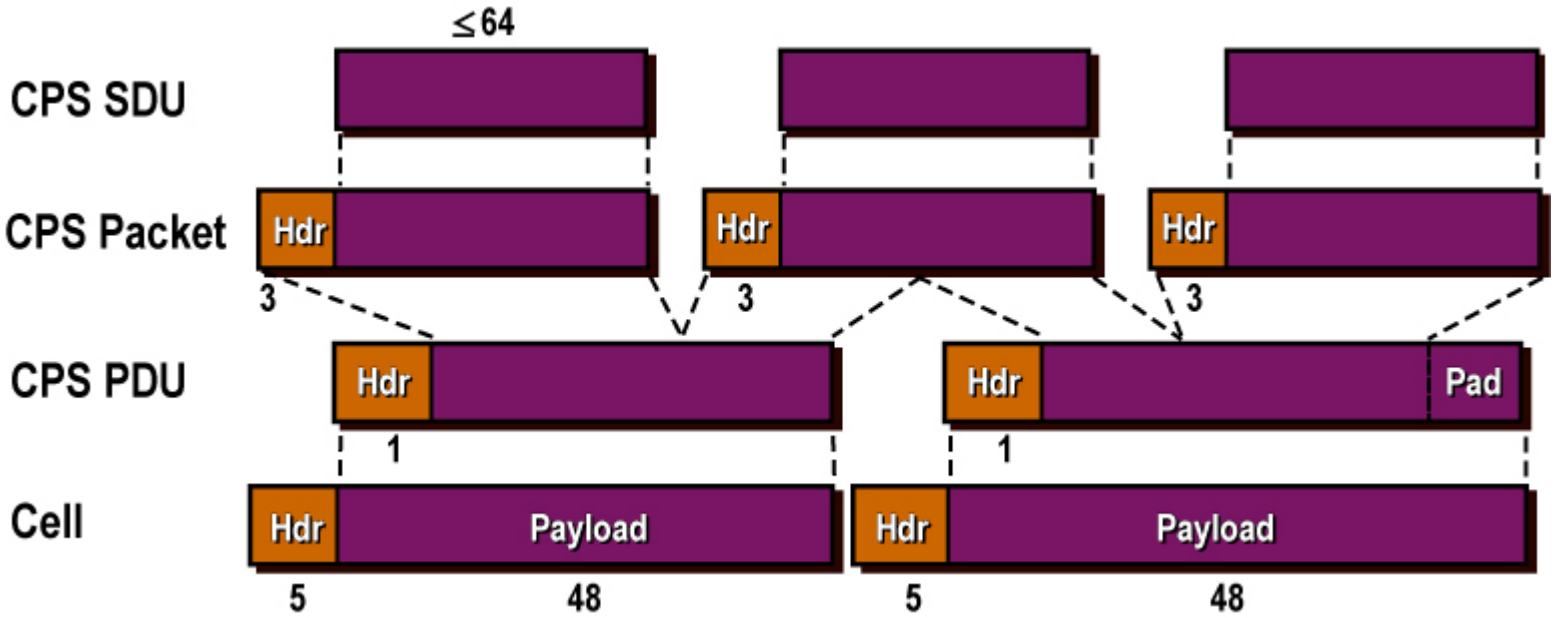
AAL2

- The CPS header is made up of four fields.
- The “channel” of CID is at the CS layer (multiple voice channels) and is different from the channel of the VCI.
- The UUI allows data to be routed to different users.
- The Length value is one less than the payload length.
- The HEC polynomial is $x^5 + x^2 + 1$



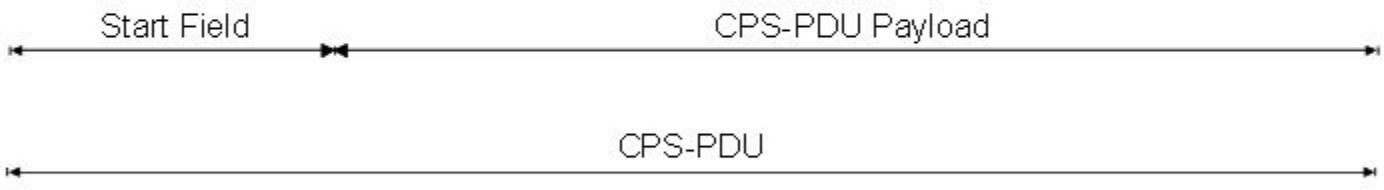
AAL2

- After the CPS packets are constructed from the CPS SDU and the header, the packets are broken down to fit into the ATM cells.
- For each cell PDU, a one octet header is added, leaving 47 octets for the CPS packet.



AAL2

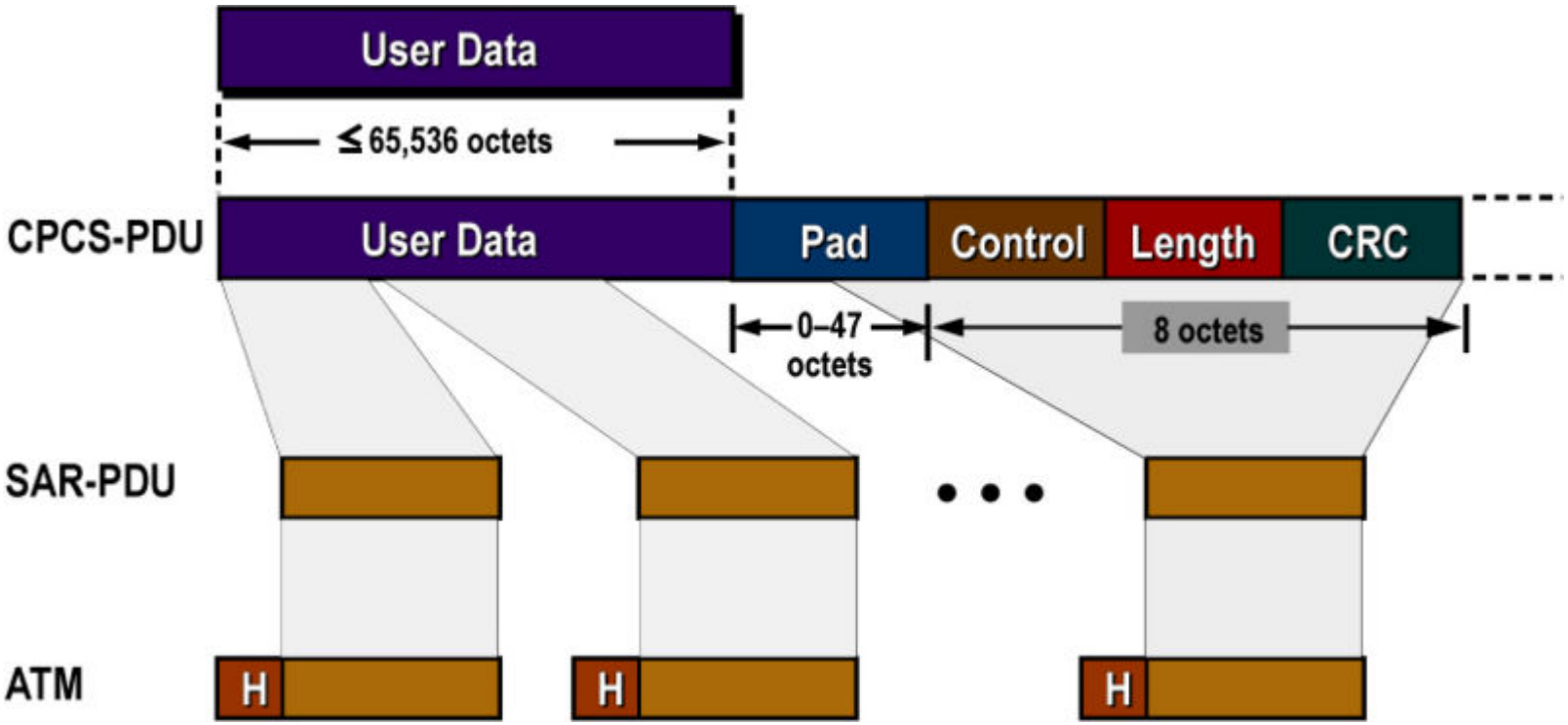
- The PDU header is made up of three fields.
- The offset field (OSF) is necessary for the CPS packet that does not align with the start of a cell.
- But a CP packet will never extend for more than 2 cells so the Sequence Number (SN) will be either 0 or 1.
- The parity bit is over the previous 7 bits.



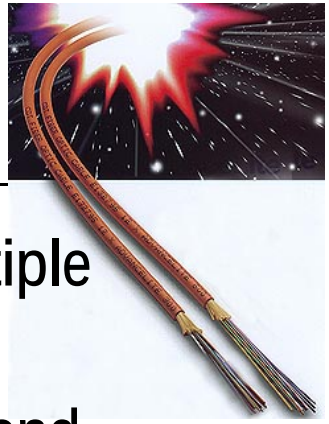
OSF Offset Field (6 bits)
SN Sequence Number (1 bit)
P Parity (1 bit)
PAD Padding (0 to 47 octets)

AAL5

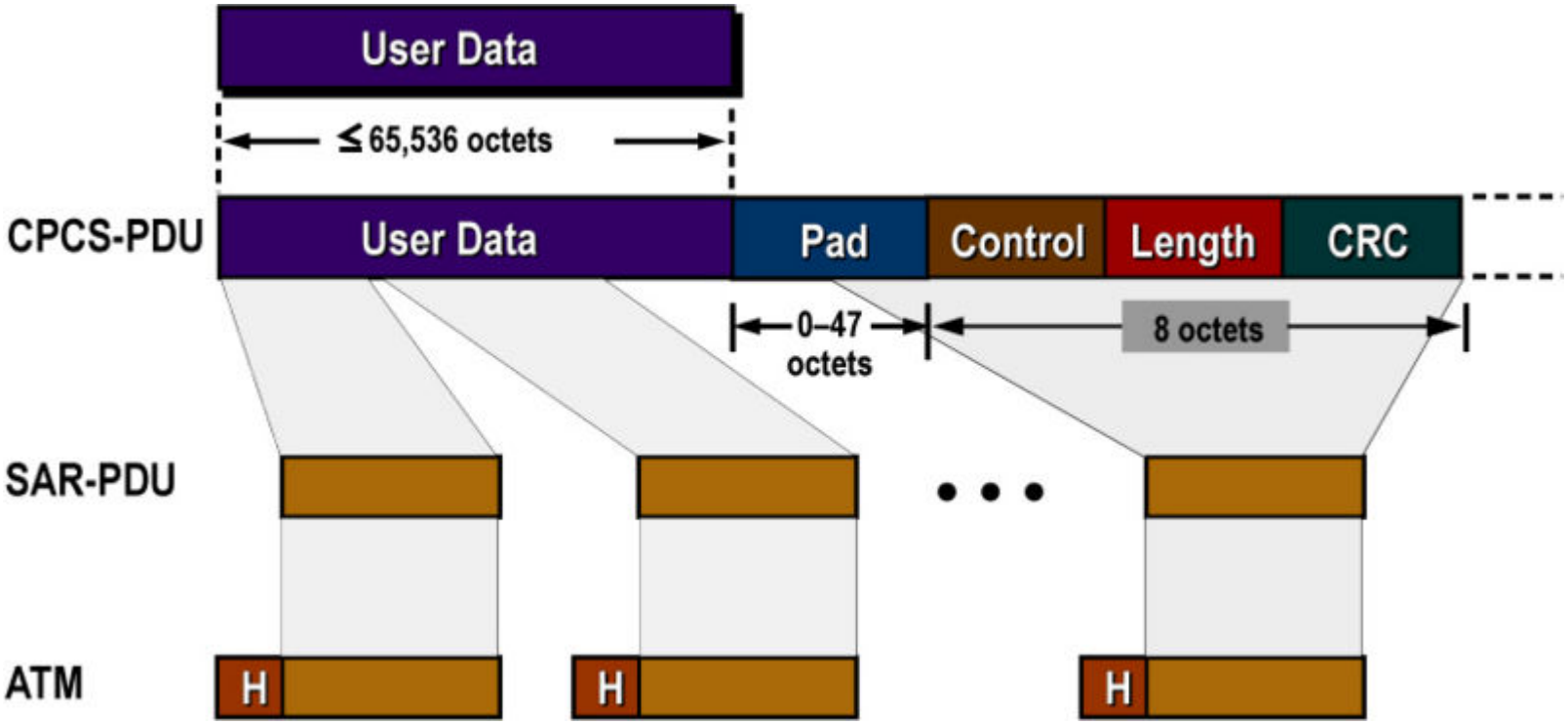
- AAL5 is a stripped down CPS function primarily intended for support of protocols such as TCP/IP.
- Basically, it takes a frame of up to 65,535 octets, appends an eight octet trailer and then breaks the result down into 48 octet PDUs.



AAL5

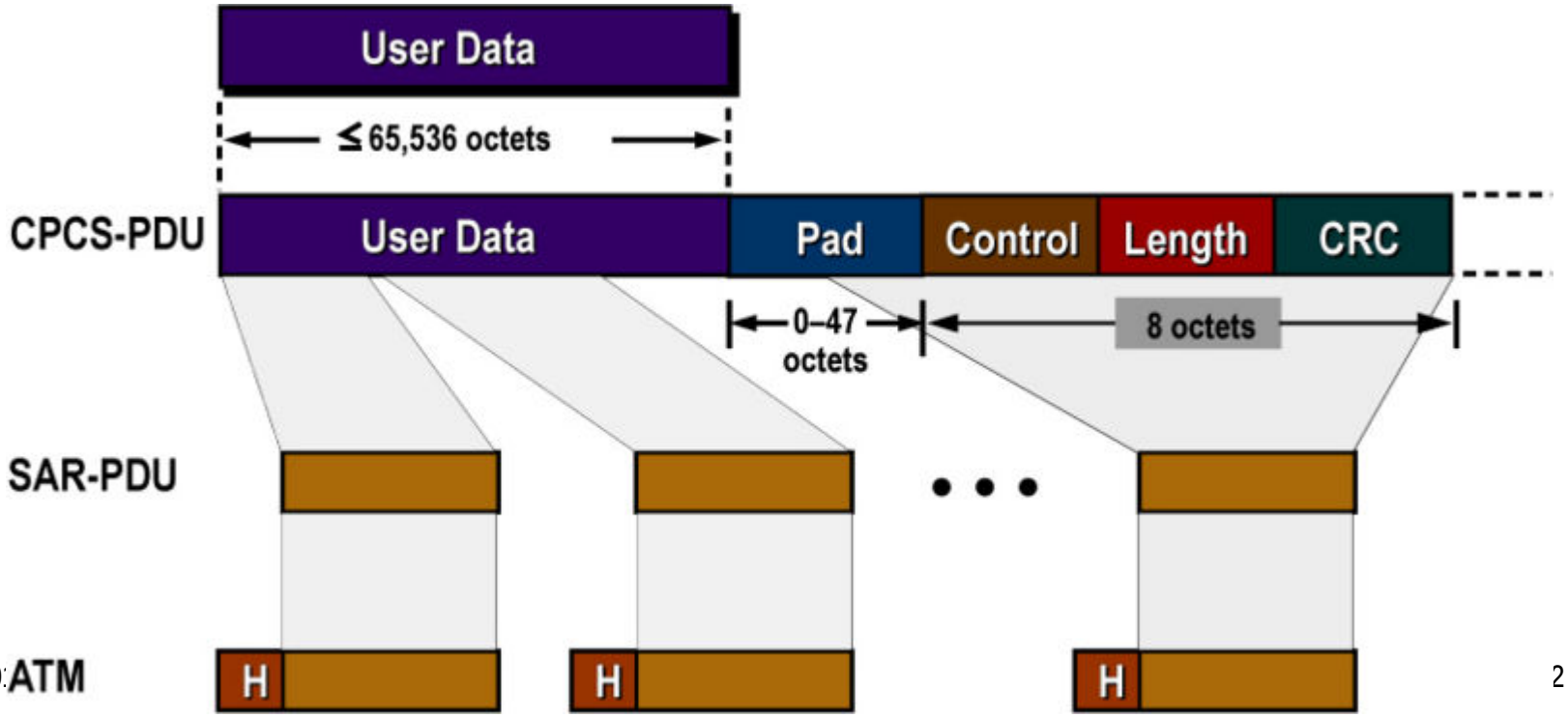


- The user data is padded so that the total length is a multiple of 48 (so it will fit "evenly" into the ATM cell PDUs).
- The Control field is 2 octets. The first is available to the end users to specify the protocol. The second octet is not used.



AAL5

- The Length field is two octets and is used to indicate the length of the user data.
- The CRC is a 32 bit (4 octet) CRC (long polynomial) .
- To indicate the end of a set of cells making up a user frame, the low order bit of the Payload Type Indicator (PTI) in the ATM header is set to 1.

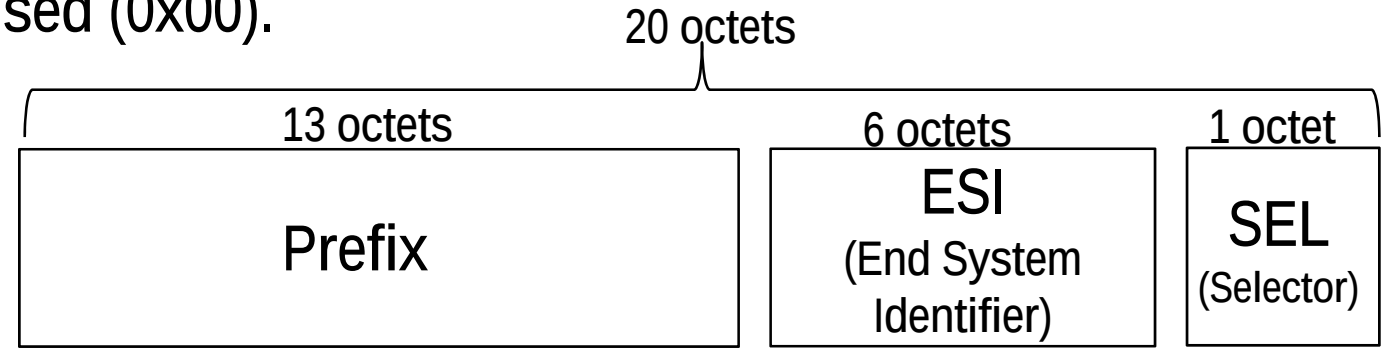


ATM Signaling

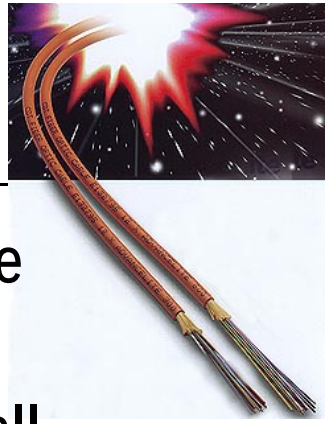
- Now that we've examined the components of an ATM system, it's time to look at the operation of the network, specifically the signaling that sets up the PVCs and SVCs.
- One of the basic concepts is that each station on an ATM network has to have a unique address, just like in TCP/IP.
- When a VC is to be set up, the calling station will provide the address of the called station, in much the same manner that you dial a telephone number to connect to another person.
- The ATM address is called the "ATM End Station Address" or AESA. We'll take a look at that now.

ATM End Station Addressing (AESA)

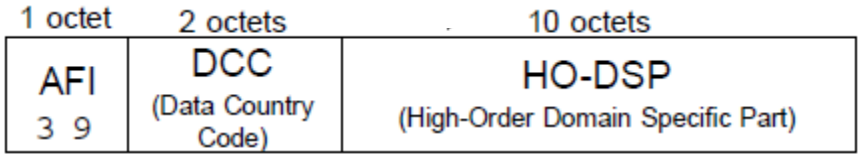
- All ATM switches and end stations must have a unique address for signaling. The overall address is 20 octets.
- The Prefix identifies the ATM switch the end device is attached to.
- The ESI identifies the end system. Within the prefix, it must unique, but better if it's globally unique. If Ethernet is used to connect to the ATM switch, the Ethernet MAC is a good choice.
- The Selector octet is for vendor use, but is generally not used (0x00).



AESA Prefix

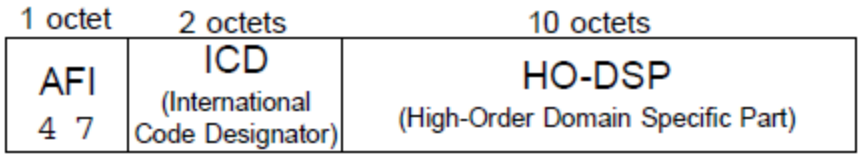


- For private ATM networks, there are three formats for the prefix.
- The Authority and Format Identifier (AFI) is common to all three. The AFI codes are defined in ITU X.213, Table A.4.
- The DCC format is similar to the ICD format but the DCC is a country code while the ICD can apply to a company.
- With the ICD format, a company can auto configure the ATM switches.



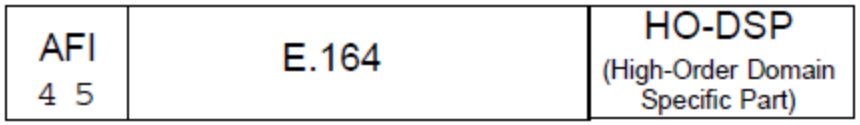
DCC AESA Format

Country codes (in BCD) as specified by ISO 3166 (For example US is coded 840). Field left justified and padded by "F"



ICD AESA Format

The ICD is similar to the DCC but is defined in ISO 6523

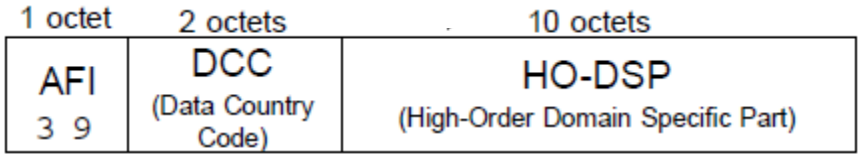


E.164 Based AESA Format

This addressing format combines the E.164 addressing scheme (used in PSTN and N-ISDN) with the AESA. The E.164 number is padded with leading zeros and by a trailing "F". For example +01(619)594-7898 would be represented by: 000016195947898F

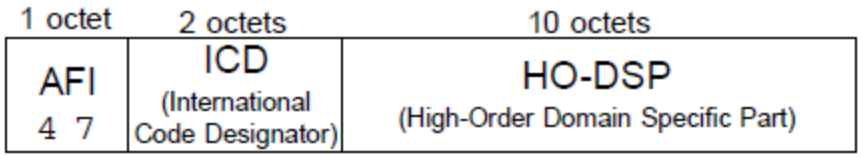
AESA Prefix

- The last format is E.164 based, which is essentially a standard telephone number format.
- A telephone number is a maximum of 15 digits. If encoded in BCD, it will take 7.5 octets. The last nibble is set to F. See example below the diagram.
- If the number is less than 15 digits, the number is right justified (with the F) and padded on the left with leading zeroes.
- Public ATM networks use the E.164 format.



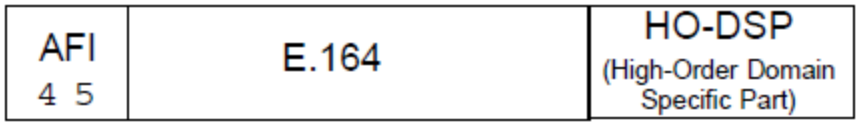
DCC AESA Format

Country codes (in BCD) as specified by ISO 3166 (For example US is coded 840). Field left justified and padded by "F"



ICD AESA Format

The ICD is similar to the DCC but is defined in ISO 6523



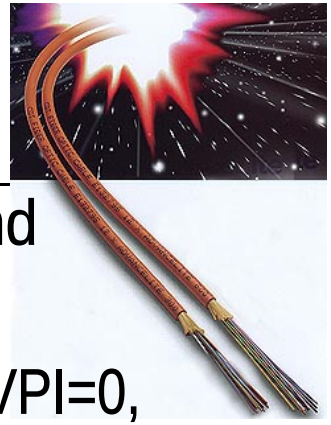
E.164 Based AESA Format

This addressing format combines the E.164 addressing scheme (used in PSTN and N-ISDN) with the AESA. The E.164 number is padded with leading zeros and by a trailing "F". For example +01(619)594-7898 would be represented by: 000016195947898F

Integrated Local Management IF (ILMI)

- I mentioned earlier that the Prefix identifies the port on the ATM switch serving the end stations. If the E.164 prefix is used, the number has to be manually programmed into the port.
- But how do we get the proper address into the end station? If this was TCP/IP, we'd used DHCP. Well, ATM has an equivalent of DHCP, called Integrated Local Management Interface (ILMI).
- When an end station is connected to the ATM port, it will ask the ATM switch for its 13 octet prefix.
- Upon receiving the prefix it will then send the full 20 octet address, consisting of the prefix, ESI and SEL, to the ATM switch.
- The switch can now route VCs to the end station.

Setting up a Virtual Channel

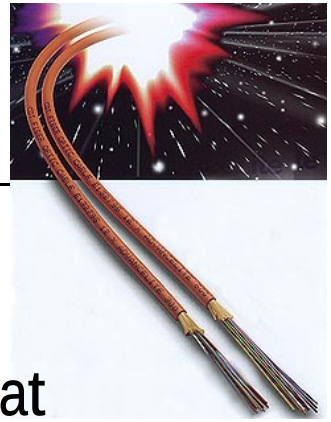


- Once the AESA address is established, a user at the end station can place a call across the network.
- Call setup requests are sent across the dedicated VC, VPI=0, VCI=5.
- I won't go through the whole sequence but here's the short form:
  - The user needs to know the AESA of the called party.
  - A request for connection is sent to the ATM network.
  - Using Dijkstra's algorithm (or equivalent), a path is found through the network to the called station.
  - The called station is sent a connection request.
  - If accepted, the called party is connected, a response is sent back, and the calling party is connected.
  - Data transfer commences.

Setting up a Virtual Channel

- During the call setup, the QoS of the call is specified and the VC chosen must be able to meet that QoS.
- The switches that the VC goes through enter the specifications of the call in their routing table.
- The routing table contains the incoming VPI (and maybe the VCI if it's a channel switch) and the outgoing VPI (VCI).
- Alternate paths are usually specified in case there's a link failure in the VC.
- All cells for that call follow the same path through the network.
- I will not go through the frame structures for the call setup.

Network-Network Interface (NNI)



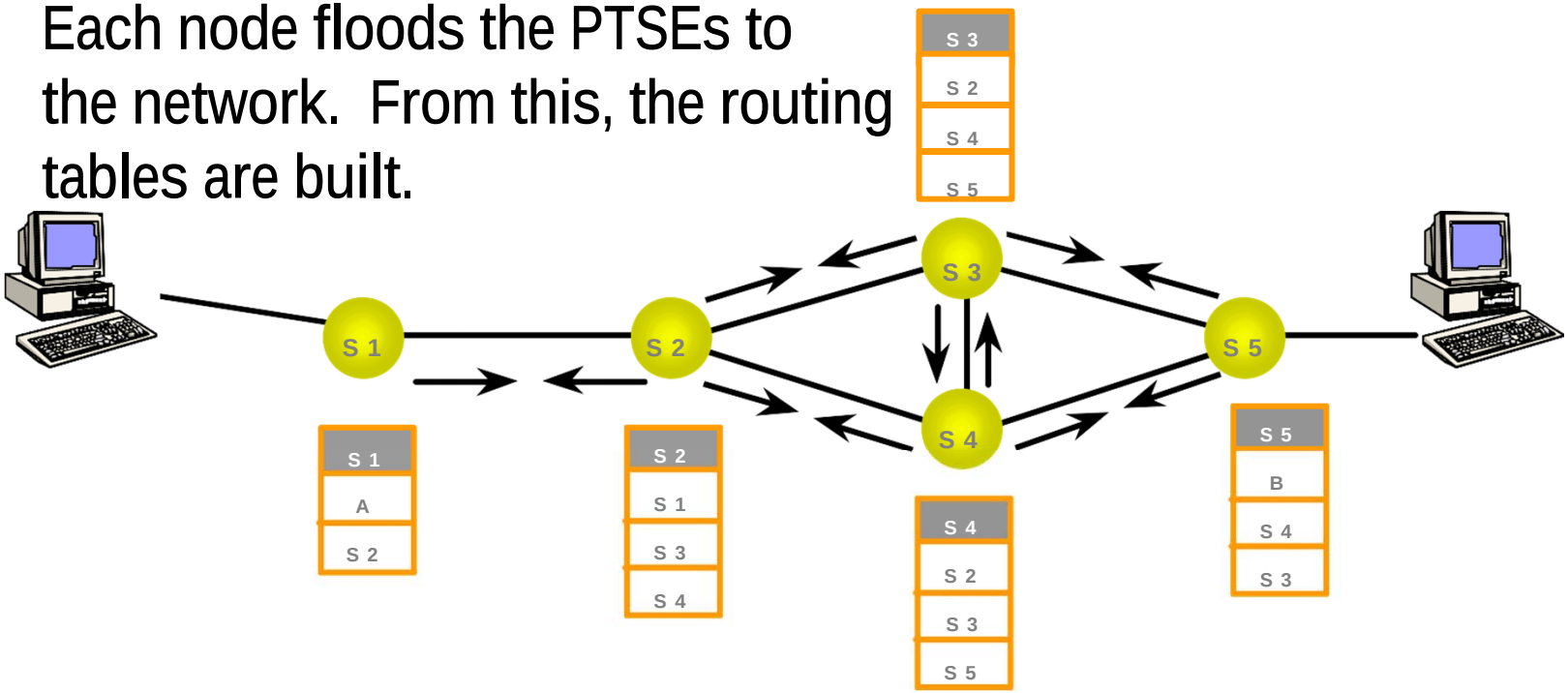
- Network-to-Network signaling consists of protocols that implement signaling between ATM switches.
- There are two routing protocols that are used to set up a route through the network.
  - Interim Interswitch Signaling Protocol (IISP)
  - Private Network to Network Interface (PNNI)
- IISP is generally not used except at border nodes in a network so I won't describe it here.

Private Network-to-Network Interface (PNNI)

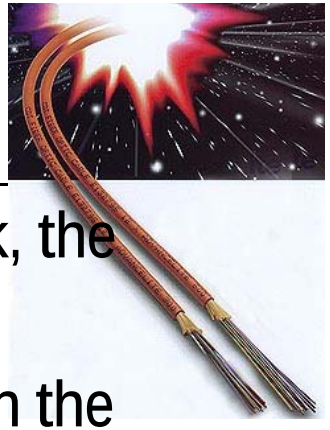
- PNNI has two parts: Routing and signaling.
- It is very complex. The specification is over 350 pages. But let's give it a try!
- First, I'll look at routing – specifically how the routing tables are built.
- The philosophy of ATM is "Route once, switch many."

PNNI - Routing

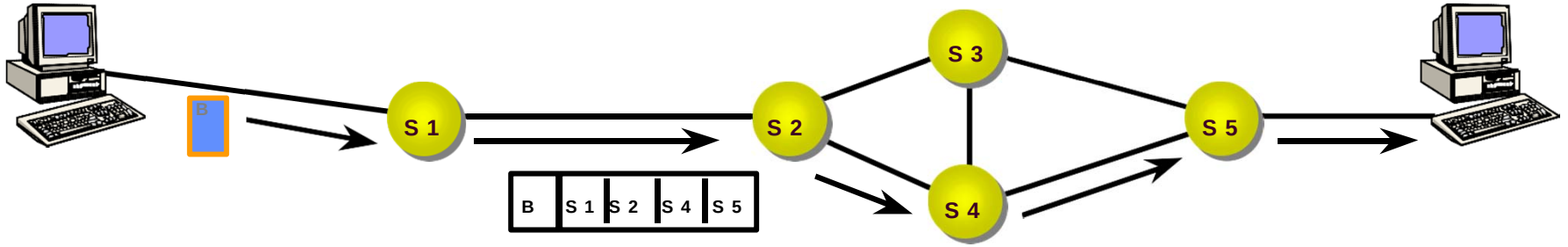
- Each node periodically exchanges “Hello” messages with its neighbors.
- Each node constructs “PNNI Topology State Elements” (PTSEs), describing the node and listing links to direct neighbors, as shown below.
- Each node floods the PTSEs to the network. From this, the routing tables are built.



PNNI – Source Routing

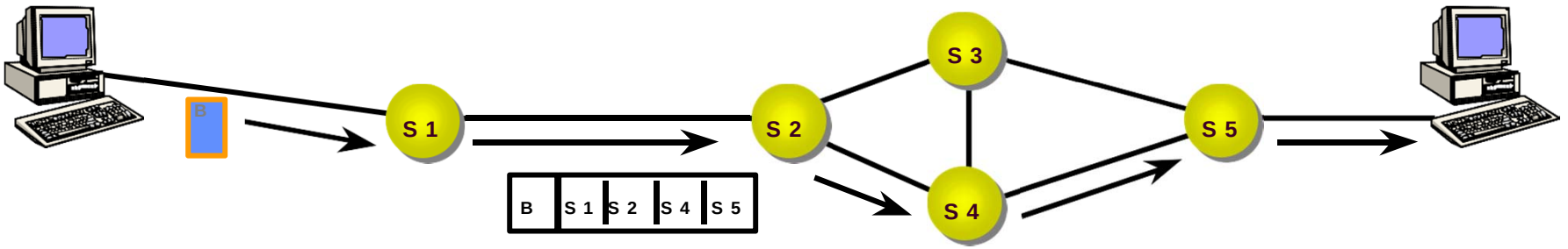


- When a user sends a connection request to the network, the ingress ATM node has the whole network topology.
- It uses a routing algorithm to find the best route through the network, given the QoS of the connection request.
- The ingress node adds the route to the connection request and the connection request flows along that route.
- Each node makes an entry for the VPI/VCI of the connection. Note that the VPI/VCI entry is for the connection from the previous node.
- The cells for that connection will then be switched, not routed.



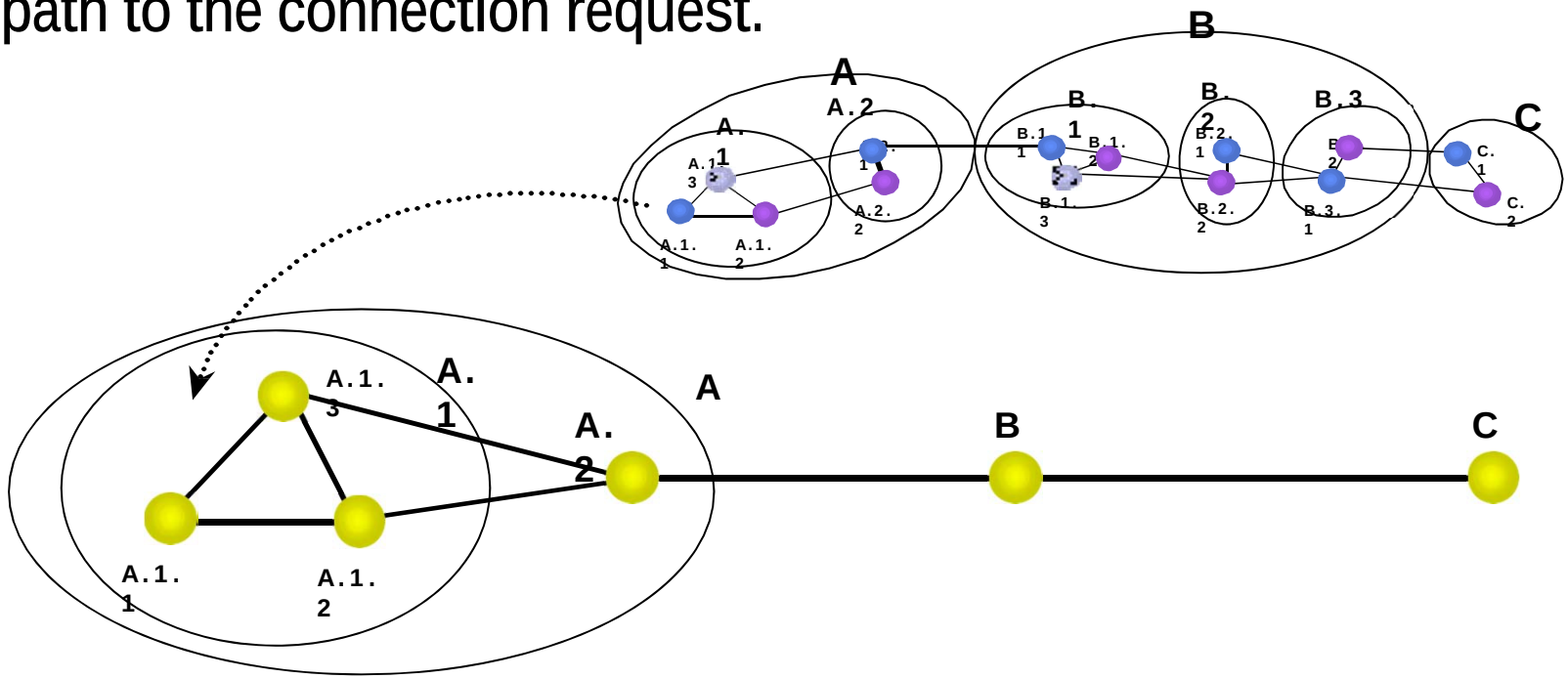
PNNI – Crankback

- If a node in the VC selected by the ingress node cannot support the QoS requested, it will reject the connection request.
- This will cause the ingress node to attempt another route. This is called “Crankback”.
- If no path is found that will support the QoS, the call fails.



PNNI – Routing

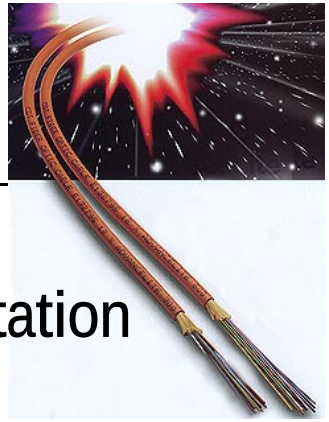
- For very large networks, PNNI will divide the network into a hierarchy of peer groups.
- Each group will know the topology of itself, but only know that the other groups are in the path of a connection.
- The ingress node of the peer group will route and add the path to the connection request.



ATM - Summary

- ATM was designed to be “all things to everyone”. During the standards development, the cry was “ATM everywhere!”
- But some of the traffic that ATM was designed to carry never materialized, especially interactive video, which left primarily voice traffic and data (especially IP).
- Because it was designed to do so much, ATM is complex, which means the nodes are more expensive than IP routers.
- While ATM was deployed in significant networks, it is now losing favor, being seen as unnecessary under IP traffic.
- It’s interesting that two of the oldest technologies, Ethernet in the LAN and TCP/IP in the WAN, both connectionless technologies, have become dominant.

Some ATM Questions



- Could I use an Ethernet connection between my end station and the ingress ATM node?
  - No. One of the basic concepts of ATM is that each virtual circuit has a QoS. Ethernet is a “best efforts” delivery service with no QoS. Therefore, ATM across other network technologies is not defined.
    If you want an ATM connection from an end station, there must be a native ATM connection to the ingress node.
    Note that the opposite is defined. Ethernet over ATM is defined in the LANE (LAN Emulation) specification. It is complex.

The Internet

- The concept of the Internet is different from the networks we've looked at so far.
  - The networks we've looked at all have native addresses and can only send data to end points with those kind of addresses, over that network (example: ATM can only send data to end devices with an AESA, and only over an ATM network).
  - Those networks were not designed to be overlaid on top of other networks.
- The Internet was designed from the beginning to operate over essentially any type of network, and on almost any type of computer. The design overlays a wide variety of heterogeneous networks and computers.

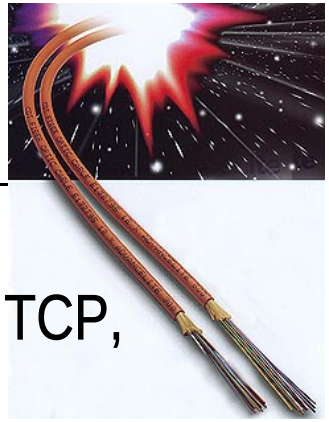
The Internet

- The term “Internet” comes from the term “Internetwork”, meaning “Inter-network”.
- The “work” portion got thrown away and it became Internet.

Agenda for this Section

- There's a lot to the Internet and many ways to approach describing how it works. What I present here is just the basics – an introduction. A lot of details are left out.
- Here's what we'll use in this presentation:
- I'll start by describing the IP addressing scheme.
- Then we'll look at the IP protocol layer and IP frame.
- Next, at how the IP layer uses hardware addresses.
- Given that much, I'll look at what happens when a computer is connected to an IP network and how it gets an IP address.

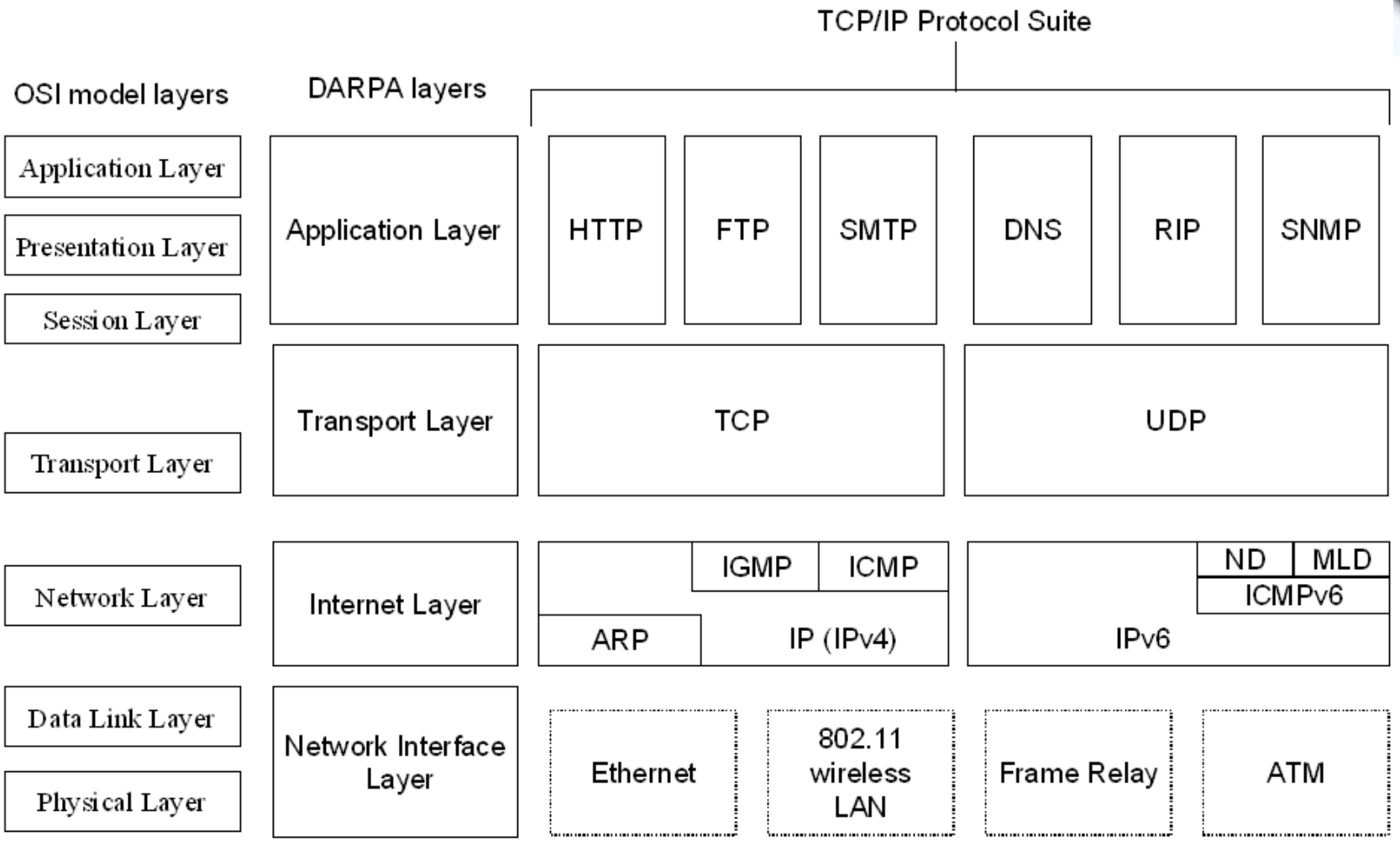
Agenda for this Section



- We'll move up the protocol stack, examining UDP and TCP, including the concept of ports.
- Next we'll look at how routing is done and the protocols needed to build the routing tables.
- Above the TCP layer are the services and we'll begin looking at those.
- The Domain Name System.
- The World Wide Web (HTTP).
- Things I'm leaving out: Electronic Mail, Internet Security (IPSec and https), NAT, VPN, RTP and probably a few other things.

The Internet Protocol Suite

- The figure below reviews the Internet layers and some of the protocols. I'll review many of these in this section



IP Addressing

- There are two versions of IP addressing now.
  - IP Version 4, or IPV4
  - IP Version 6, or IPV6
- I'll cover IPV4 first, then IPV6.
- Question: What happened to IP Version 5? Why did we jump from IPV4 to IPV6?

IPV4 Notation

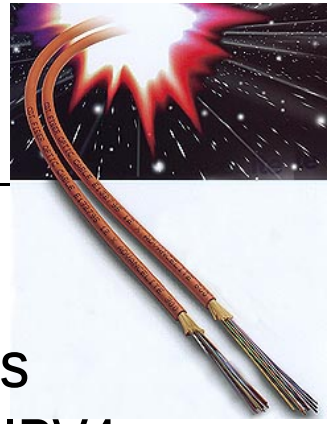
- IPV4 addresses are 32 bits long. How can we represent these addresses in an easy to read and remember form?
- The technique chosen was to take the 32 bit address 8 bits at a time and use the decimal value of those eight bits.
- So the binary address

  10000000 00001010 00000010 00011110

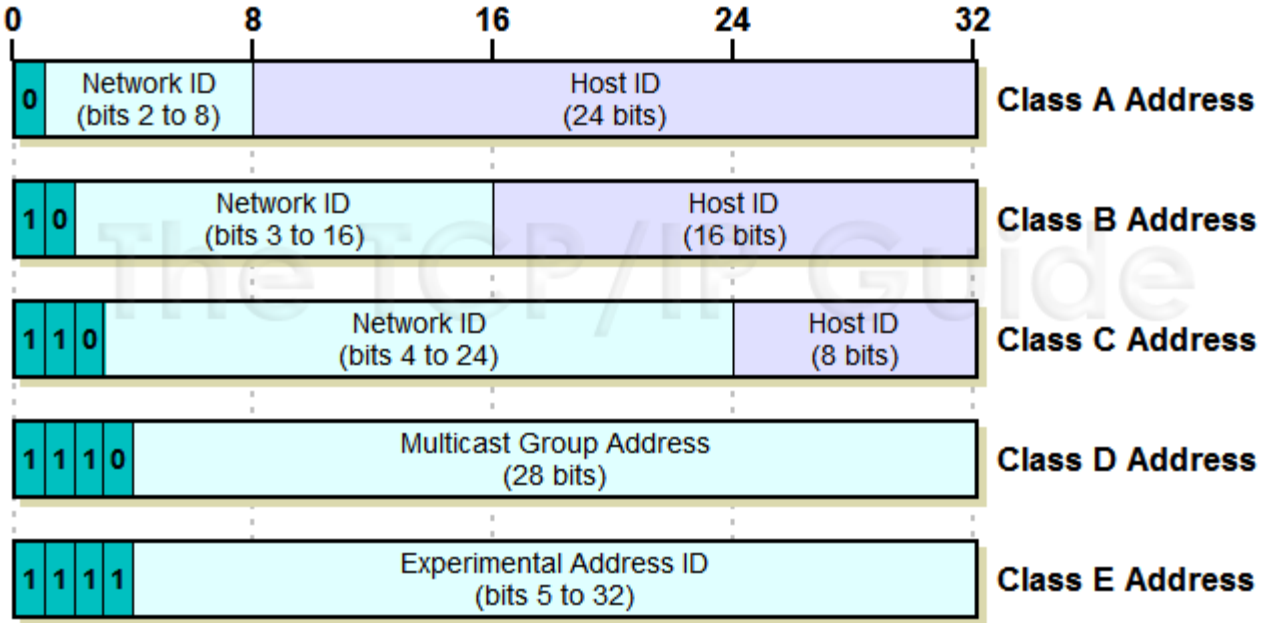
  will be represented as

  128.10.2.30
- This is known as “dotted decimal notation”.

IPV4

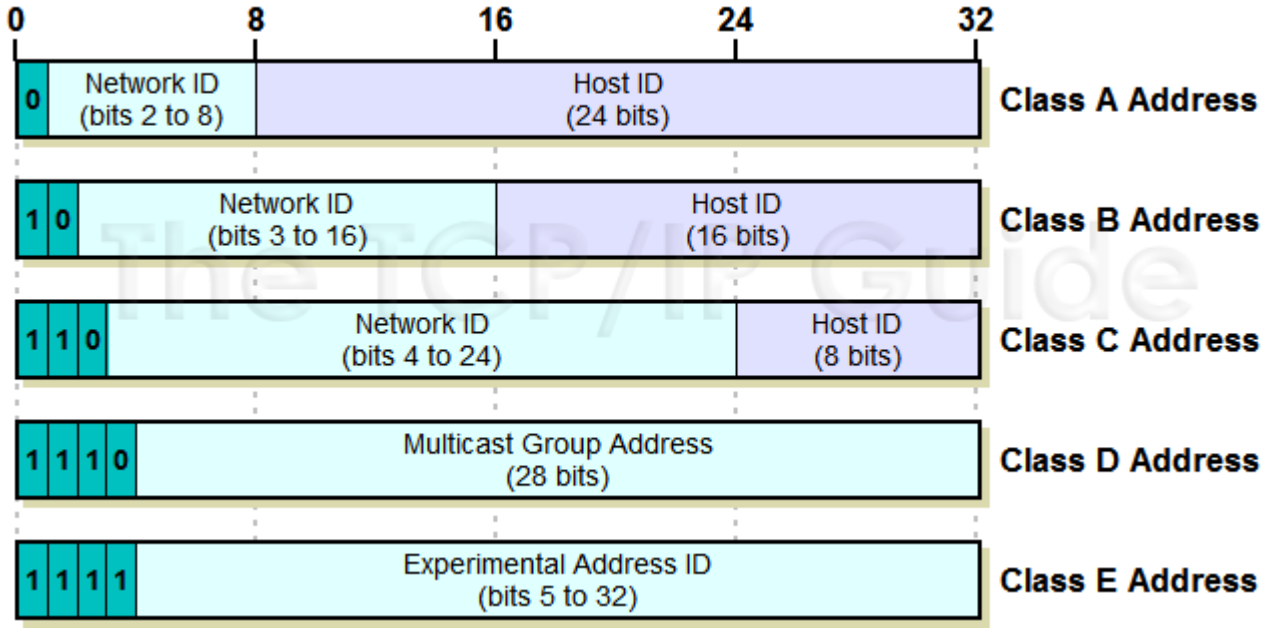


- IPV4 has gone through some revisions in its history. Initially, it was “classful” addressing, but the designers changed it to “classless” address to extend the life of IPV4.
- I’ll start with the classful addressing.
- All IPV4 addresses are 32 bits in length.



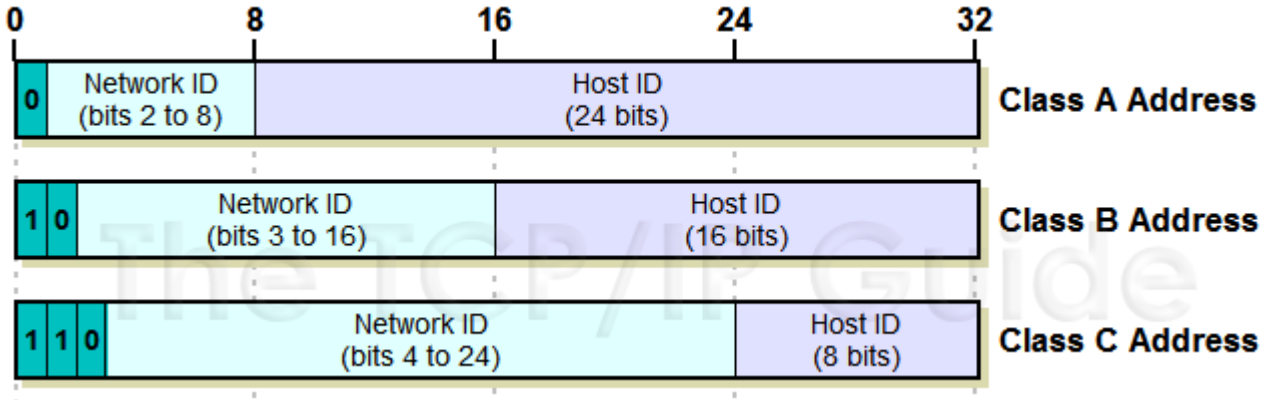
IPV4 Classful Addressing

- Five different types of addresses were defined, although we're only going to look at the first three.
- Note that the different classes are identified by the initial bits of the address.



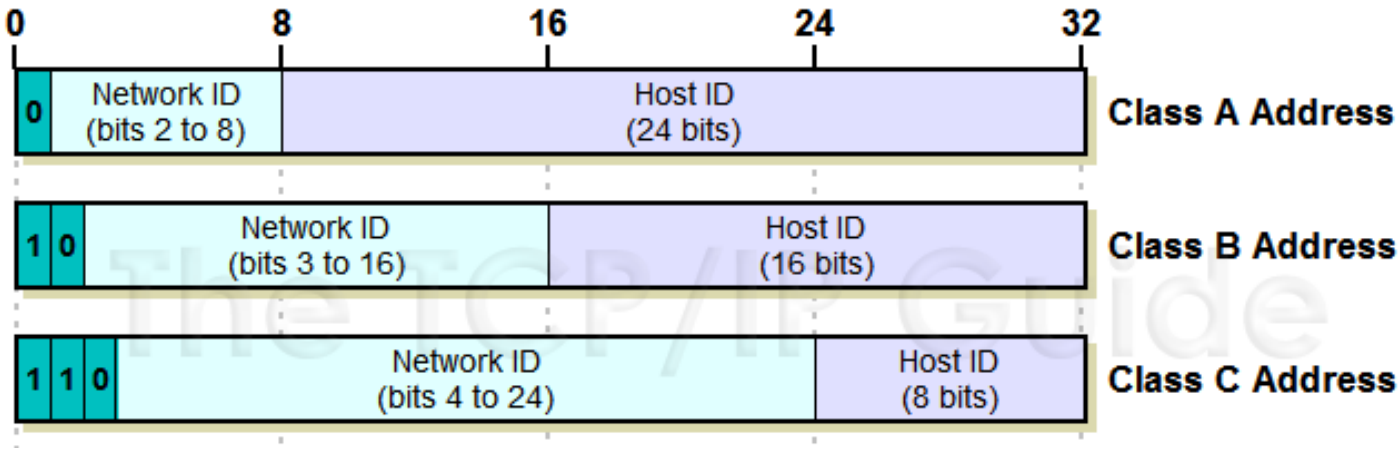
IPV4 Classful Addressing

- The IPV4 address was divided into two parts, a Network ID part and a Host ID part.
- The thought was that there would be a small number of networks that needed a lot of host addresses, more networks that needed a medium number of host addresses, and a lot of networks that would need a small number of host addresses.
- Since the first bit of the Network ID for a Class A address is taken, this leaves 127 Network IDs, and 16 million Host IDs



IPV4 Classful Addressing

- For Class B addresses, 14 bits are available to the Network IDs and 16 bits for Host IDs. This gives a bit over 16,000 Networks and a few more than 65,000 Host IDs.
- For Class C addresses, 21 bits are allocated for Network IDs and only 8 for Host IDs. This gives over 2 million Network IDs but only 254 Host IDs.
- This division between Network IDs and Host IDs was not very efficient.



IPV4 Classful Addressing

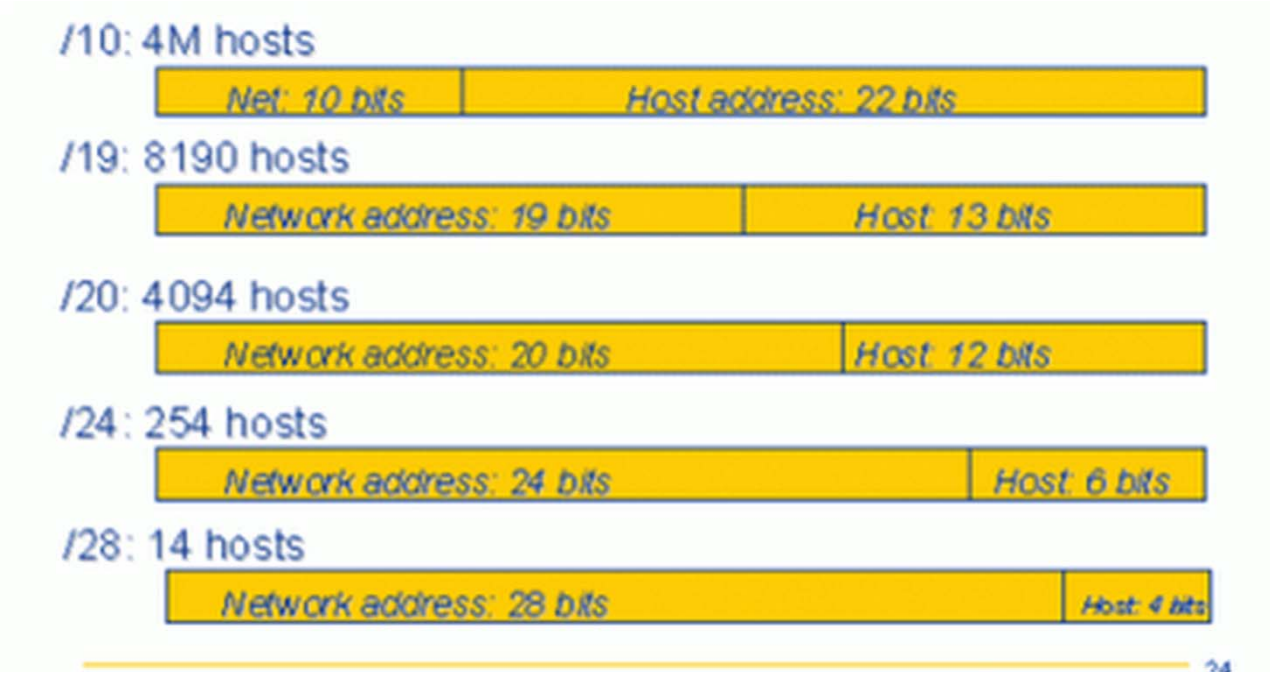
- This breakdown of the IP address space was leading to the exhaustion of addresses.
- Almost no one needed 24 bits for Host IDs. (Class A addresses)
- Class C addresses were essentially useless because most companies needed more than 254 Host IDs
- Class B addresses were rapidly being exhausted because there were only a bit more than 16,000 possible Network IDs.
- To extend the life of IPV4, Classless addressing was developed.

IPV4 Classless Addressing

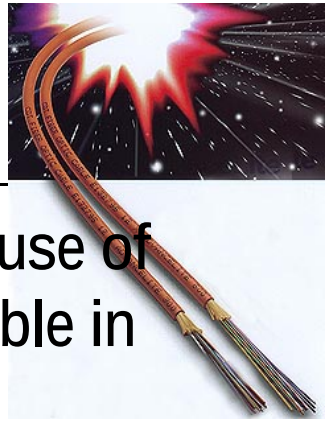
- Classless Inter-Domain Routing (CIDR) was initially published (RFC 1518 & 1519) in 1993.
- It did away with classful addressing and made the addressing scheme more flexible.
- The difference was that the Network ID portion of the address was made variable. So a Network ID could be 15 bits, or 20 bits, etc.
- These addresses are displayed with a slash (/) and the number of bits in the Network ID portion of the address.
- Example: 128.211.168.27 /21 This indicates that the first 21 bits are the Network ID.

Examples of Classless Addresses

- Here are some more examples of classless IPV4 addresses.



IPV4 Private Addresses



- Another technique that has saved IP addresses is the use of Private IP Addresses. Private addresses are not routable in the global Internet.
- Companies use these private addresses within the company and use Network Address Translation (NAT – discussed later) to communicate with the global Internet.
- The private address are:
  - 10.0.0.0 /8
  - 172.16.0.0 /12
  - 192.168.0.0 /16
  - 169.254.0.0 /16
- The 192.168.0.0 /16 address is used in essentially all home network routers.

IPV6 Addressing

- IPV6 is defined in RFC 2460, released in 1998.
- An IPV6 address is 128 bits long (16 octets). Normally, it is divided into 64 bits for the Network ID and 64 bits for the Host ID.
- $\mathbf{2^{64}}$ is a very large number, so large that it's essentially impossible that we would run out of Network IDs, or that any network would ever have that many Hosts.

Representation of IPV6 Addresses

- Since IPV6 addresses are 16 octets long, representing them in dotted decimal notation can result in a very long string.
- Example:
  104.230.140.100.255.255.255.255.0.0.17.128.150.10.255.255
- To make address representation more compact, the designers created the “colon hexadecimal notation”, usually called “colon hex”.
- In colon hex, each four bits are represented by the hex character for it’s value and the hex values are grouped into 2 octet groups.
- Example: (same value as above)
  68E6:8C64:FFFF:FFFF:0:1180:96A:FFFF

IPV6 Notation

- It is expected that, because of the very large address space, many IPV6 addresses will have a string of zeroes.
- To make the IPV6 address representation more compact, a technique known as “zero compression” was developed.
- Let’s take the address FF05:0:0:0:0:0:0:B3
- With zero compression, this address can be represented as FF05::B3
- This works because we know that the address has 16 octets and we can fill in the appropriate number of zeroes.
- Zero compression can only be applied once in an address.
- IPV6 also uses the CDIR notation. Example: 12AB::CD30:0:0:0:0 /60 specifies that the first 60 bits are the Network ID.

"Special" IPV6 Addresses

- "Private" addresses exist in IPV6, called Unique Local addresses (ULA). Any address starting with FD, e.g., FDxx:xxxx:xxxx… is private.
  - Private addresses are discouraged in IPV6. The recommendation is that every device have it's own unique, routable address.
- Any address starting with FF is a multicast address. The next two nibbles define the type of mulitcast. Example: FFx2::/16 is a link-local multicast.
- RFC 5156 describes many of the IPV6 special addresses.
- Interesting note: Just like telephone numbers have Hollywood (fictitious) numbers, IPV6 has a Hollywood (fictitious) IP address. Anything with the prefix 2001:db8::/32 is non-routable and is intended for documentation and movies.

Special IPV6 Addresses

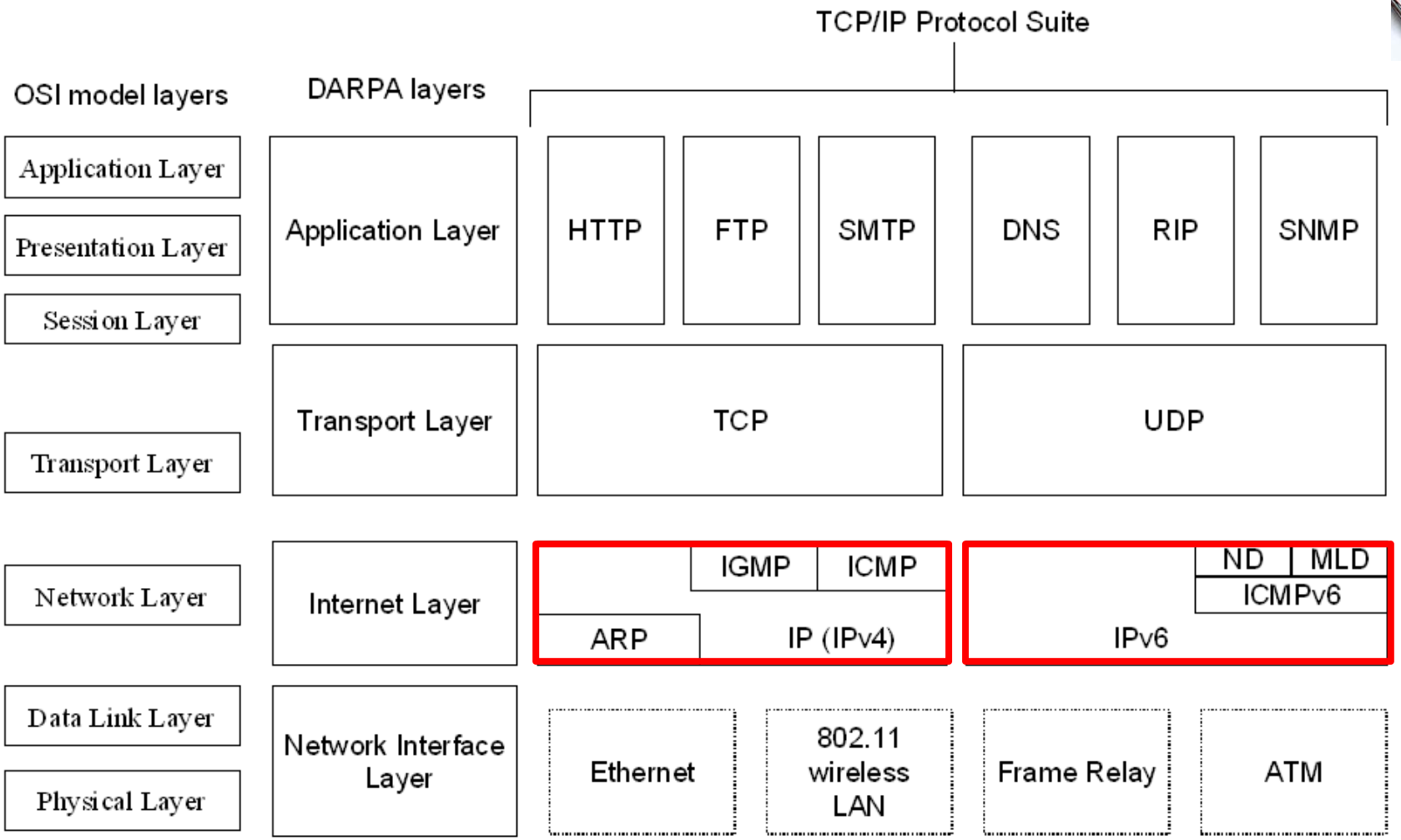
- 2000::/3 Global Unicast [RFC4291]
- FC00::/7 Unique Local Unicast [RFC4193]. This is the FDxx::/7 private address from the previous page. The 8th bit is required to set to 1.
- FE80::/10 Link Local Unicast [RFC4291]
- FF00::/8 Multicast [RFC4291]. Addresses within this group are used for autoconfiguration:
  - FF02:0:0:0:0:0:0:1 All Nodes Address
  - FF02:0:0:0:0:0:0:2 All Routers Address
  - FF02:0:0:0:0:0:0:FB mDNSv6
  - FF02:0:0:0:0:0:1:2 All-dhcp-agents
  - FF05:0:0:0:0:0:1:3 All-dhcp-servers

IP Protocol Layer

- Note that the IP (Internet) layer is independent of the Transport layer and the Network access layer.
- This will make the conversion to IPV6 easier because the higher layers of the protocol stack will not have to be changed, nor the Network access layer.
- IP is an unreliable, connectionless, best-efforts delivery mechanism.
- By unreliable we mean that delivery is not guaranteed.
- By connectionless we mean that each packet is treated independently and may take a different route from other packets.
- By best-efforts we mean that packets will be discarded only when resources are unavailable.

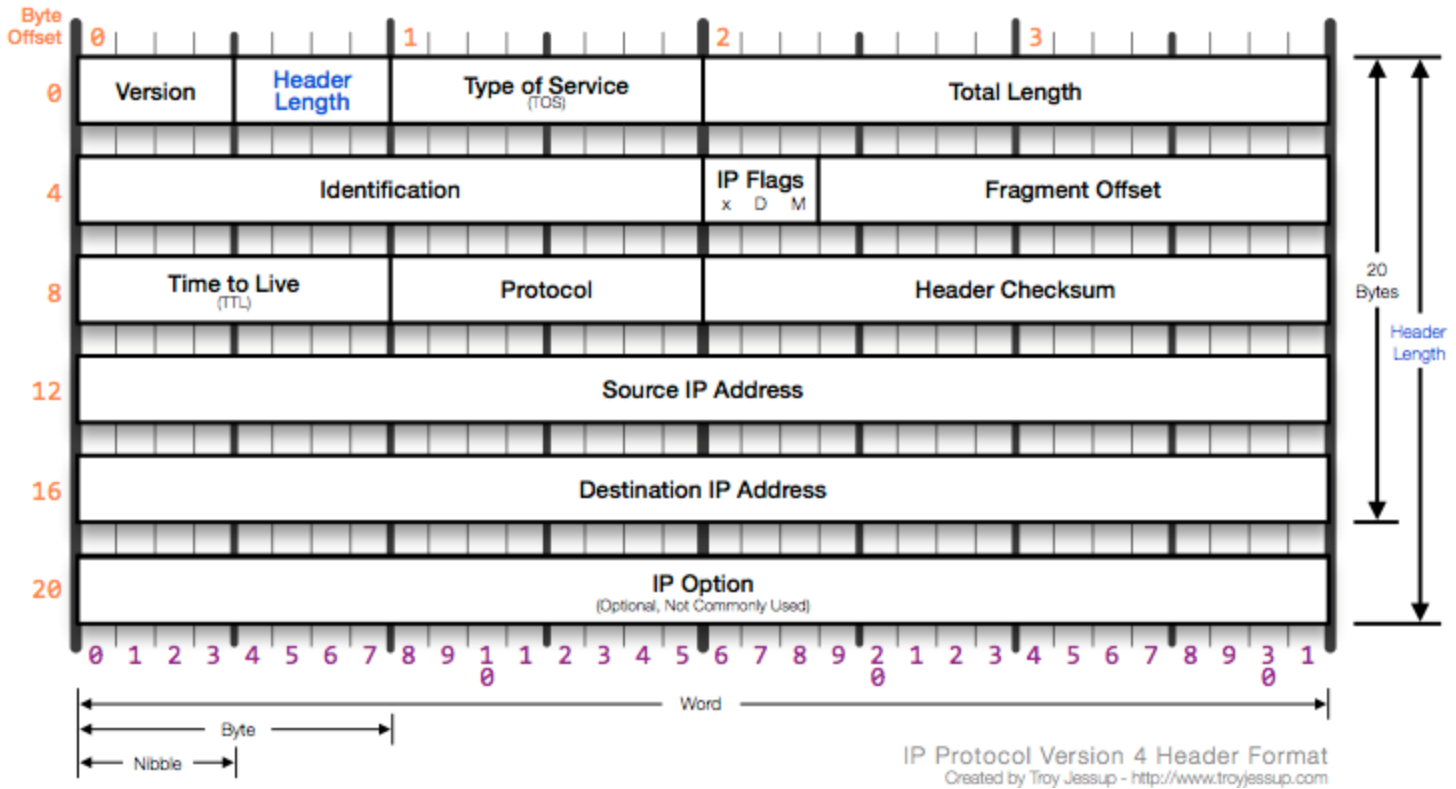
IP in the Protocol Layers

- IP is a Layer 3 protocol.



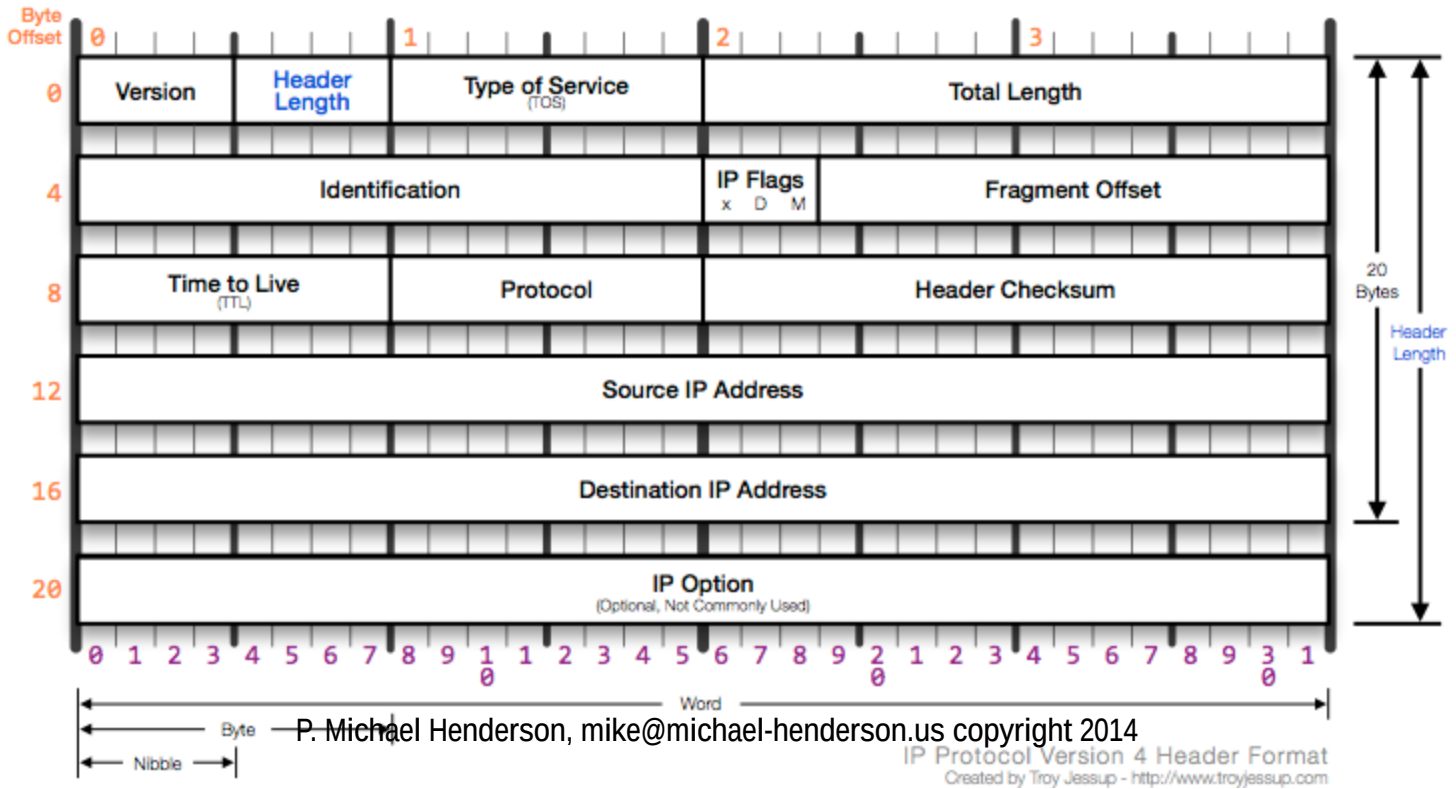
IPV4 Frame

- The IP header is in front of the payload. Note that there are no framing techniques on an IP packet. It must be framed by a lower level protocol.
- The version indicates IPV4 or IPV6. This header is IPV4.



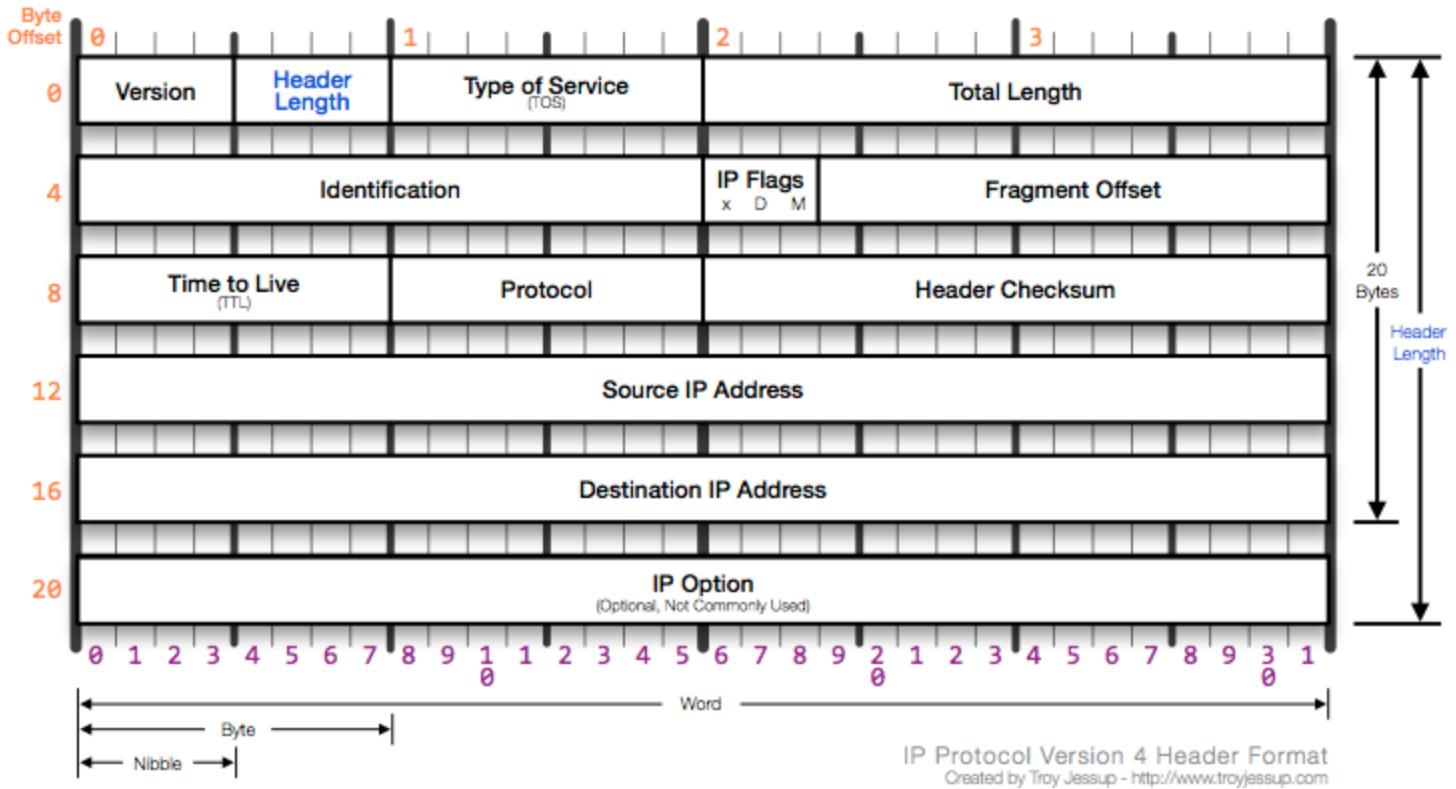
IPV4 Frame

- Time to live is a counter that is decremented by one each time the packet goes through a router. When zero, the packet is discarded.
- Protocol indicates the protocol carried. 6 is TCP, 17 is UDP. There are 142 protocol # assigned – Wikipedia has a list.



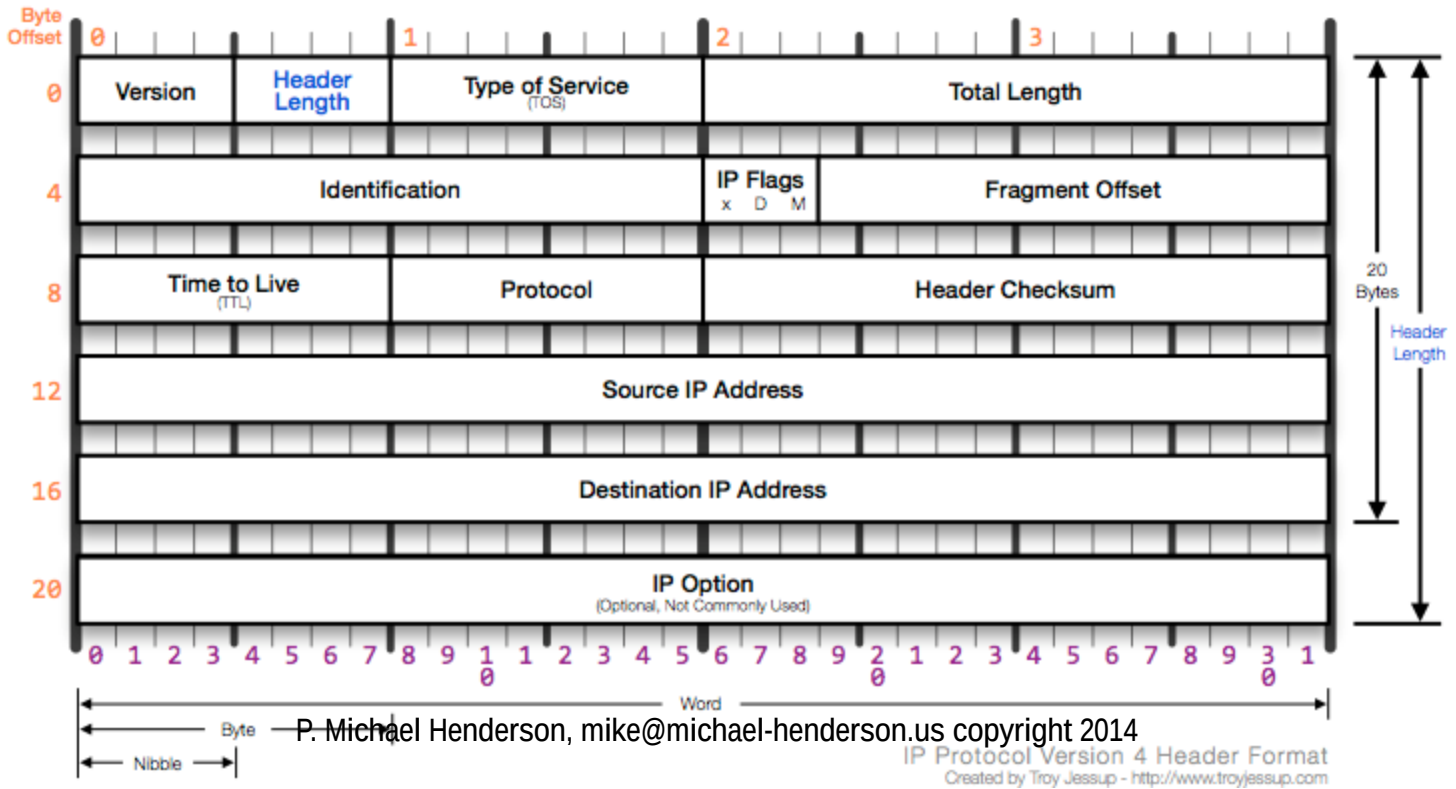
IPV4 Frame

- Identification is used to identify a group of fragments of a datagram.
- Flags are 3 bits. The first is not used. The second is “don’t fragment” if set, and the third is “more fragments”.
- The fragment offset is the fragment offset into the datagram.



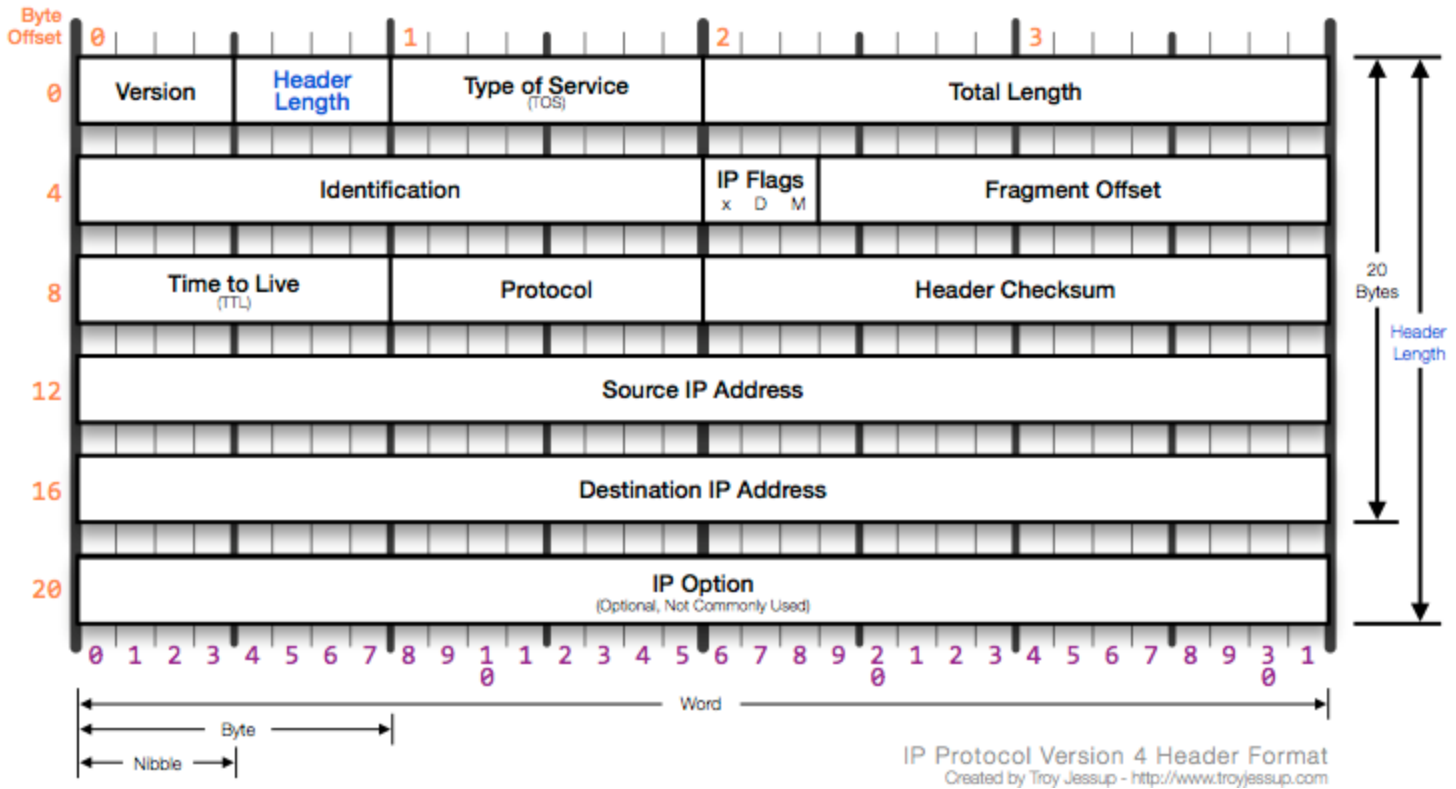
IPV4 Frame

- Time to live is a counter that is decremented by one each time the packet goes through a router. When zero, the packet is discarded.
- Protocol indicates the protocol carried. 6 is TCP, 17 is UDP. There are 142 protocol # assigned – Wikipedia has a list.



IPV4 Frame

- The Header Checksum is used to check for bit errors in the header (complement of 1s complement addition – see next slides). Note: this only covers the header, not the data.
- The Source and Destination IP address contains the IP address of the sender and the host to which the packet is addressed.



1s Complement Arithmetic

- In 1s complement arithmetic, the first bit of the word indicates whether the number is positive (0) or negative (1).
- Positive numbers are represented as ordinary binary numbers. So +18 is 00010010
- Negative numbers are represented as the 1s complement of the positive number, So -18 is 11101101
- Zero has two representations, as all 0s and as all 1s (positive and negative zero).
- If we were to add +18 and -18, we'd get all 1s (negative zero).
- When we add two numbers that cause a "carryout", such as -2 (1101) and -4 (1011), we get 11000, which is 5 bits. We take the carryout and add it back. So 1000 + 1 = 1001 (+6 is 0110 so the 1s complement, -6, is 1001).

IP Checksum

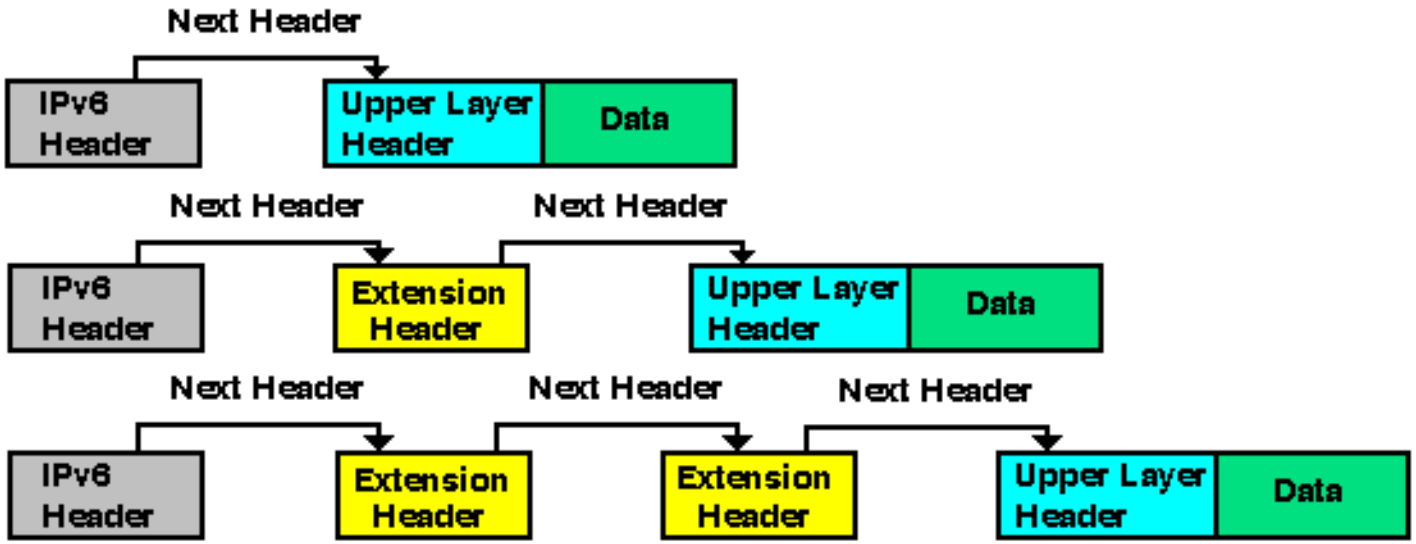
- Now, we look at the IP checksum which takes each 16 bits (padding the last word with 0s if not 16) and adds them as if they were 1s complement numbers.
- The result is then complemented. Why do we complement the result?
- Let's take our simple case, where we add -2 and -4 to get the result 1001. If that were the sum of the words in IP, we'd next complement it to 0110.
- The receiver will do a 1s complement addition and include the checksum.
- So it will add -2 and -4 and get 1001. Then it will add the checksum of 0110 which will give us a result of all 1s.
- <u>At the receiver, the result of the checksum computation will always be all 1s if the data was received without error.</u>

IP Checksum

- One special case. If the sender computes a checksum of all zeroes (after complementing), it will replace the zeroes with all ones (negative zero). This will give an all 1s result at the receiver if there are no errors.
- Incidentally, a 16 bit 1s complement checksum is a decent check on the integrity of the data. It's not as good as a well chosen CRC-16 but it's a lot less resource intensive to generate and check.
- It's more effective if used over a small number of bits, as is done with the IP header. As the number of bits covered increases, its effectiveness decreases (some bit errors are not detected).

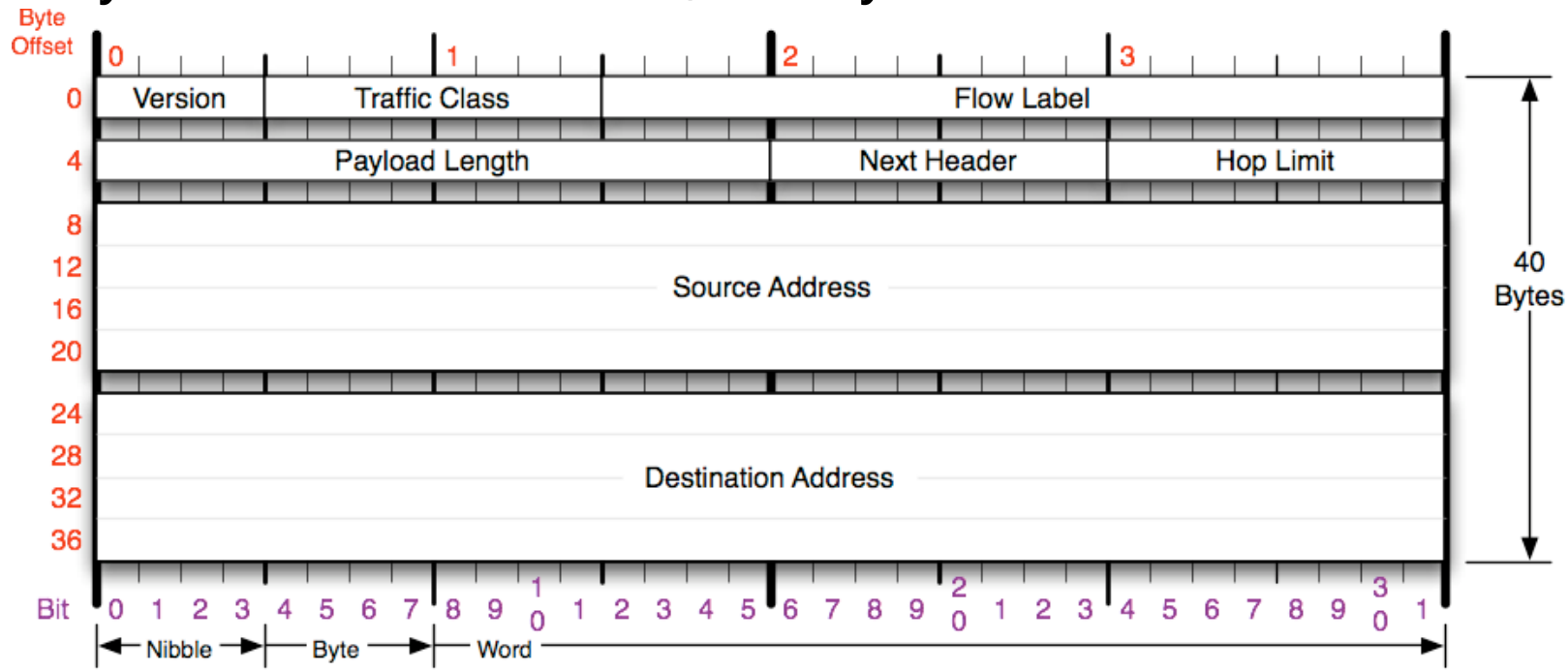
IPV6 Frame

- The approach taken for the IPV6 header is very different.
- Rather than try to put all options in one header, the designers used the concept of “extension header”.
- A datagram may, or may not, have multiple headers.
- If it has extension headers, they are required to be in a certain order, with headers pertaining to the routers first. These headers are known as “hop-by-hop” headers.



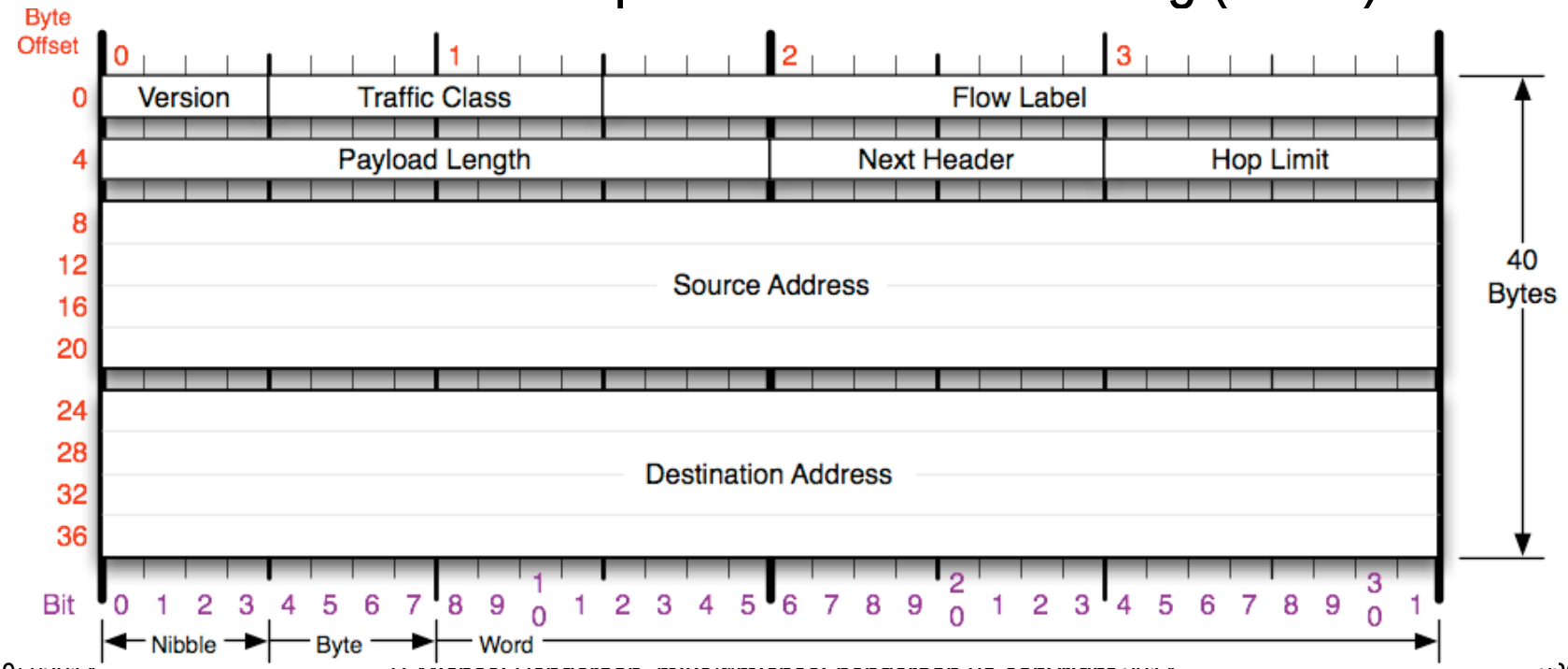
IPV6 Base Frame

- I'm only going to describe the IPV6 base frame. You'll have to look up the format of the extension frames.
- The Version field is the same as IPV4 so that a router can tell which header it is.
- Note that there is no Checksum. IPV6 depends on the lower layer to validate the frame, usually with a CRC.

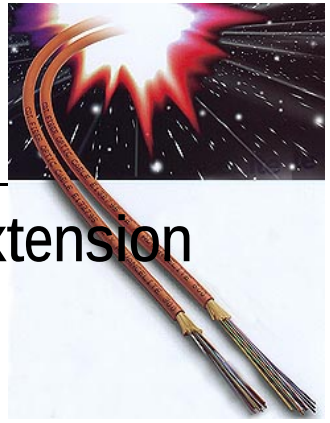


IPV6 Base Frame

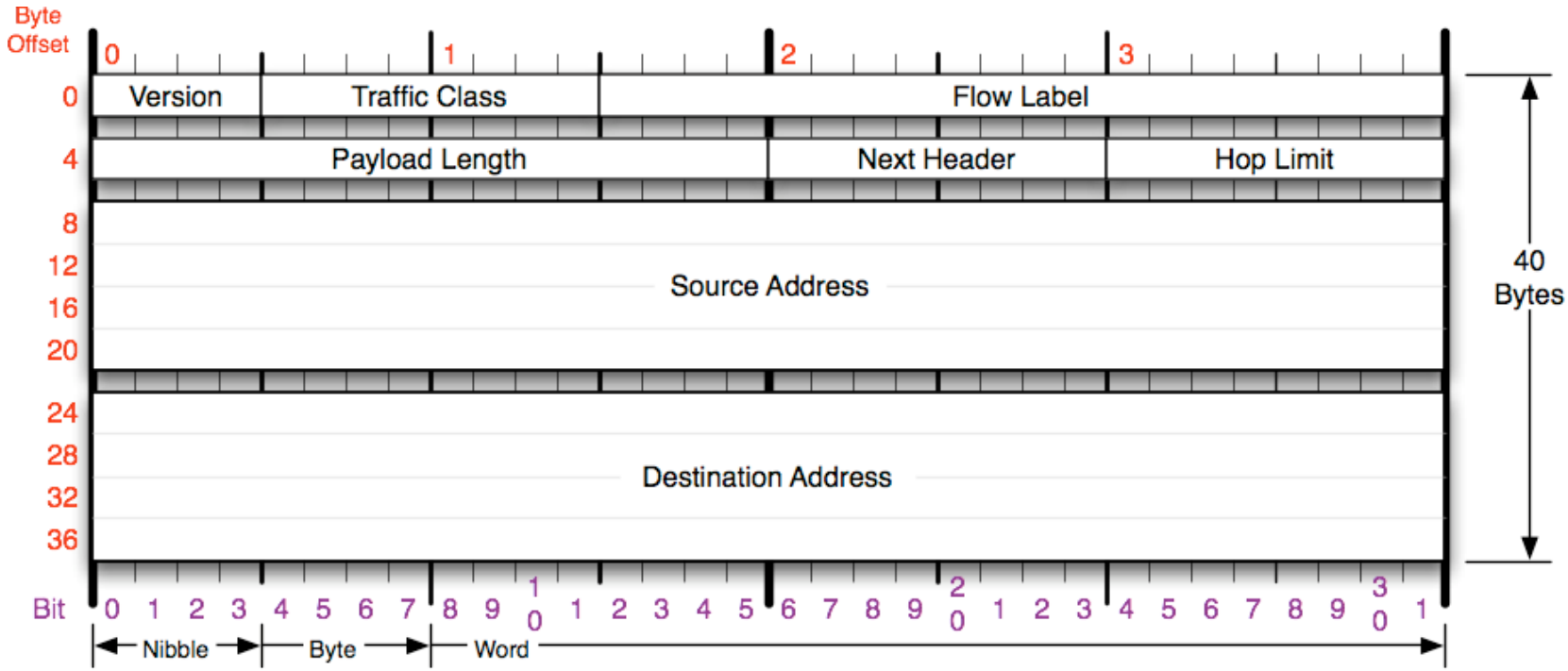
- The “Traffic Class” is the same as IPV4, Differentiated Services Code Point (DSCP).
- The Flow Label is intended to be used to direct routers to keep the packets of a specific “flow” on the same path so they do not arrive out of sequence. We’ll examine this more when we look at Multiprotocol Label Switching (MPLS).



IPV6 Base Frame



- Payload Length is the length of the payload and any extension headers.
- Next Header is similar to the Protocol field in IPV4 and indicates the type of the next header , e.g., TCP.
- Hop Limit is a number and is decremented by 1 at each node.

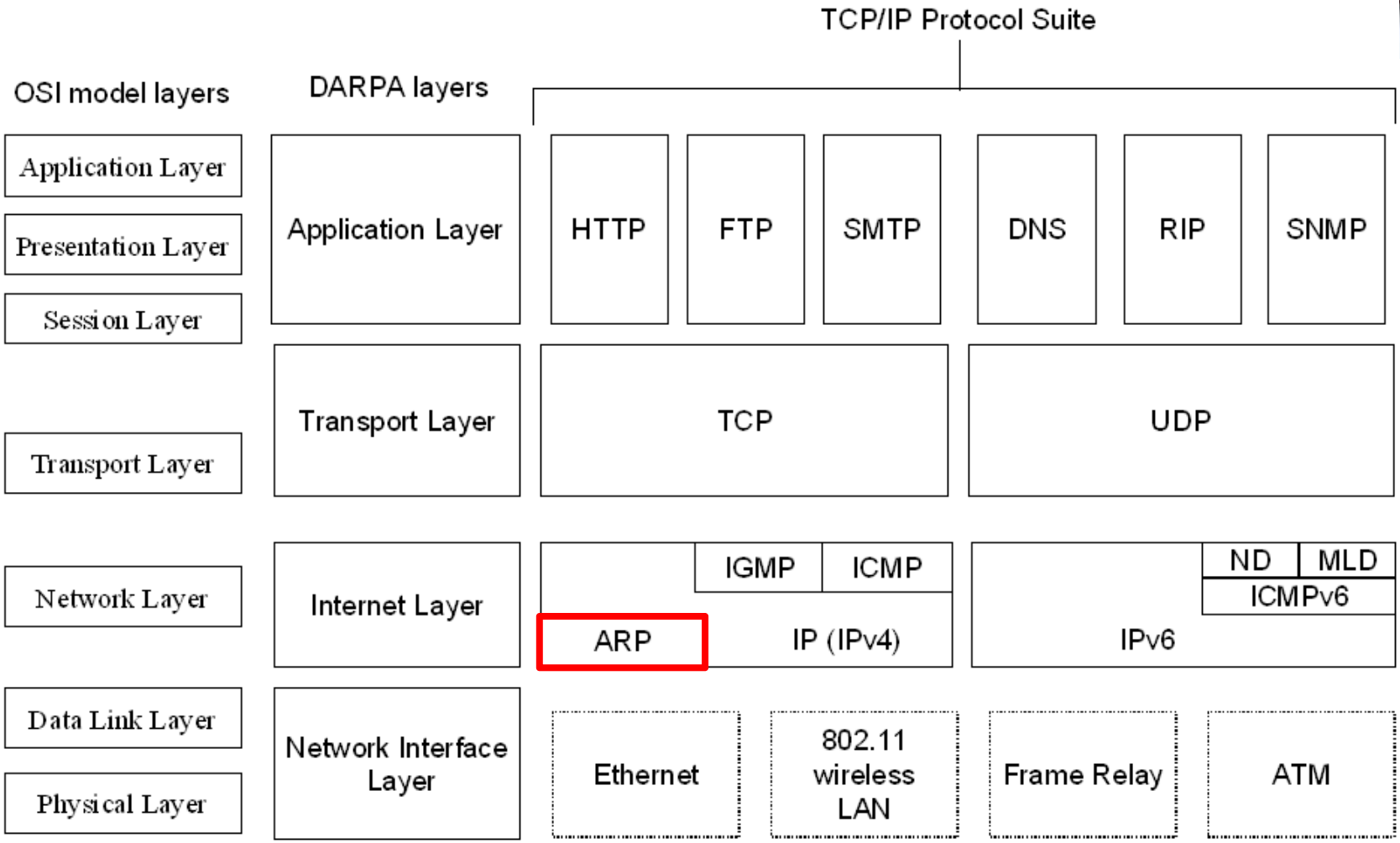


Address Resolution Protocol (ARP)

- I mentioned earlier that the Internet is a “virtual network” dependent upon other networks to carry its packets.
- So when IP packets are sent across a LAN, for example, the communication must use the hardware addresses, not IP addresses.
- How can the IP layer map IP addresses to hardware addresses?
- The answer for IPV4 is the Address Resolution Protocol (ARP).
- IPV6 uses another technique and we’ll look at that after ARP.

ARP in the Protocol Layers

- As seen here, ARP is a layer 3 protocol.



Address Resolution Protocol (ARP)

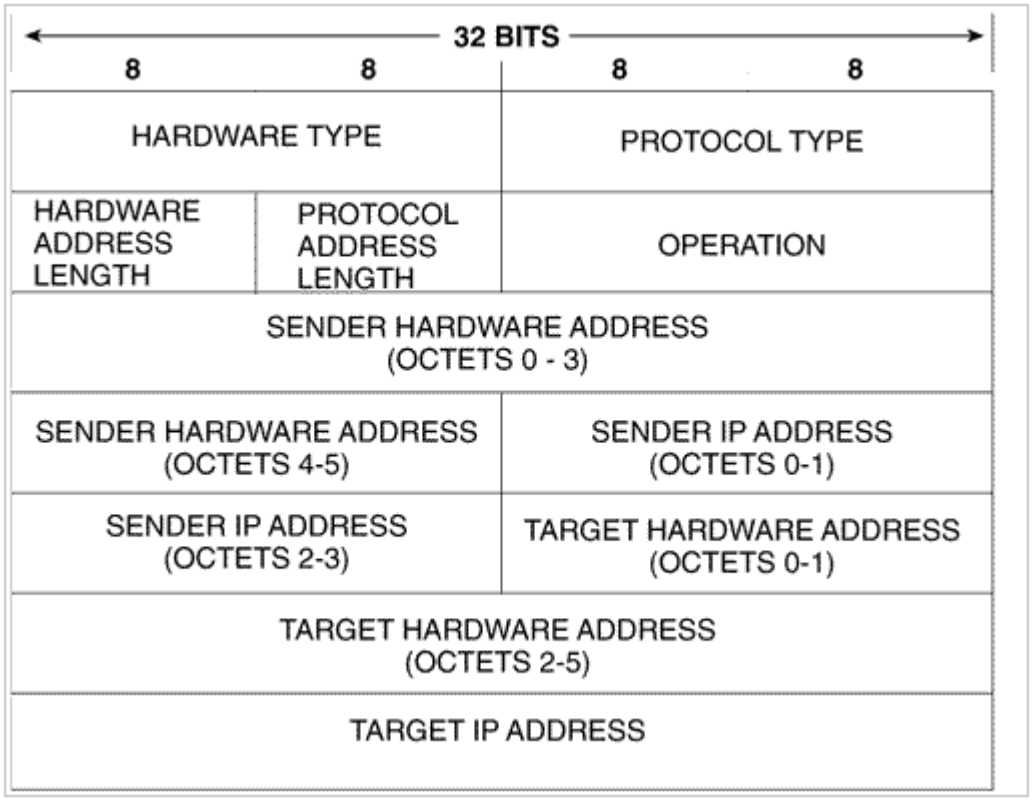
- ARP requires a broadcast physical medium, such as an Ethernet LAN.
- When one station on the LAN needs to send information to another station on the LAN, it sends out a broadcast message with its IP address, its physical address, and the IP address of the destination station.
- Basically, it's saying "Would the station with this IP address send me it's physical address".
- Since it's a broadcast, all stations receive the packet, but only the station with the requested IP address responds with it's physical address.
- After that, the sending station will use the physical address to communicate with the target station.

Address Resolution Protocol (ARP)

- You may ask, “Why not just broadcast the actual data packet to the IP address and let the appropriate target station respond?” “Wouldn’t the response have the end station’s physical address?”
- The problem is that the physical address is actually hidden by the protocol layers. The Network Address layer sends the IP packet up one layer but does not send the physical address. Therefore, all communication would have to be via broadcast which would put an unreasonable burden on all the other stations.
- With ARP, the target station puts its physical address in the ARP packet, which is sent up the protocol layers.

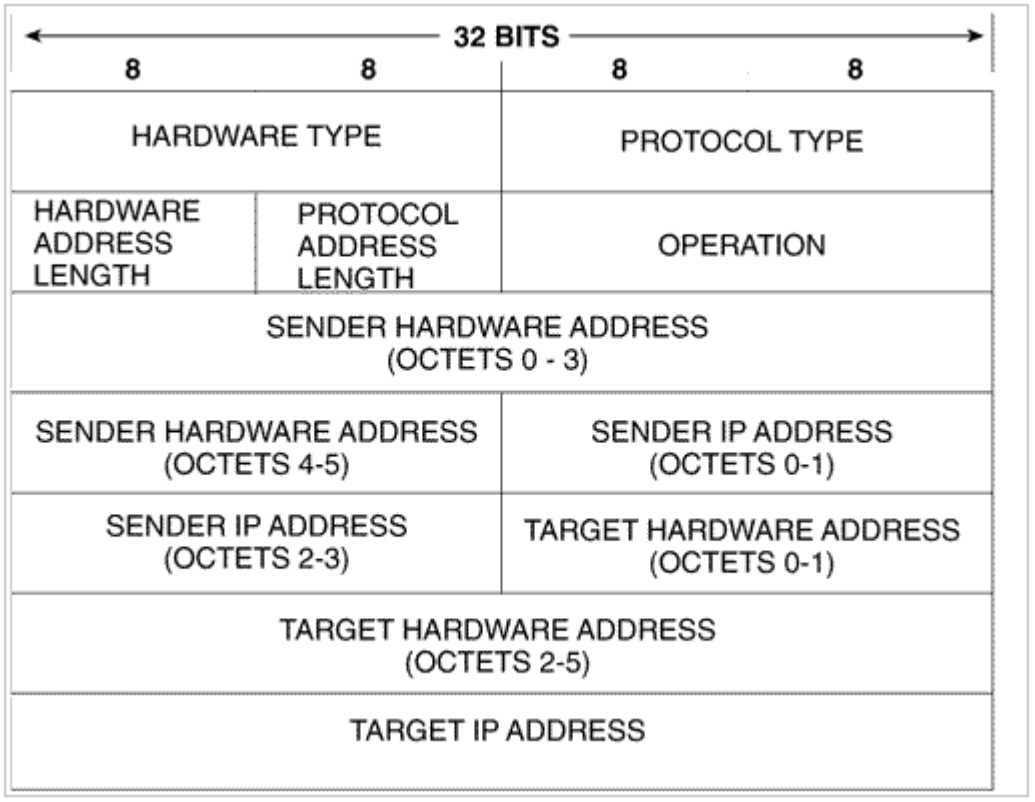
The ARP Packet

- Hardware Type specifies the physical network. Ethernet is 1.
- Protocol Type specifies the protocol the ARP request is for. For IPV4, this is 0x0800.
- Hardware Address Length is in octets. Ethernet is 6.



The ARP Packet

- Protocol Address Length is the length of the protocol address in octets. IPV4 is 4.
- Operation is type of packet. 1 is request, 2 is response.
- The addresses are self-explanatory.



Address Resolution Protocol (ARP)

- There's a lot more to ARP than described here. Those interested can read the details.
- For example, other stations listen to the traffic and build their own tables of IP to physical addresses. How long to keep that information "current" is an issue and is addressed in the standard.

ARP

- Since ARP is a Layer 3 protocol, it is not encapsulated in an IP packet. It is a Layer 3 packet in it's own right.
- When the ARP packet is sent, it is encapsulated in the lower level frame, and the type field is set in that frame to a unique code to indicate the ARP packet.
- For Ethernet, the type is 0x0806.

So How Does IPV6 Bind Addresses?

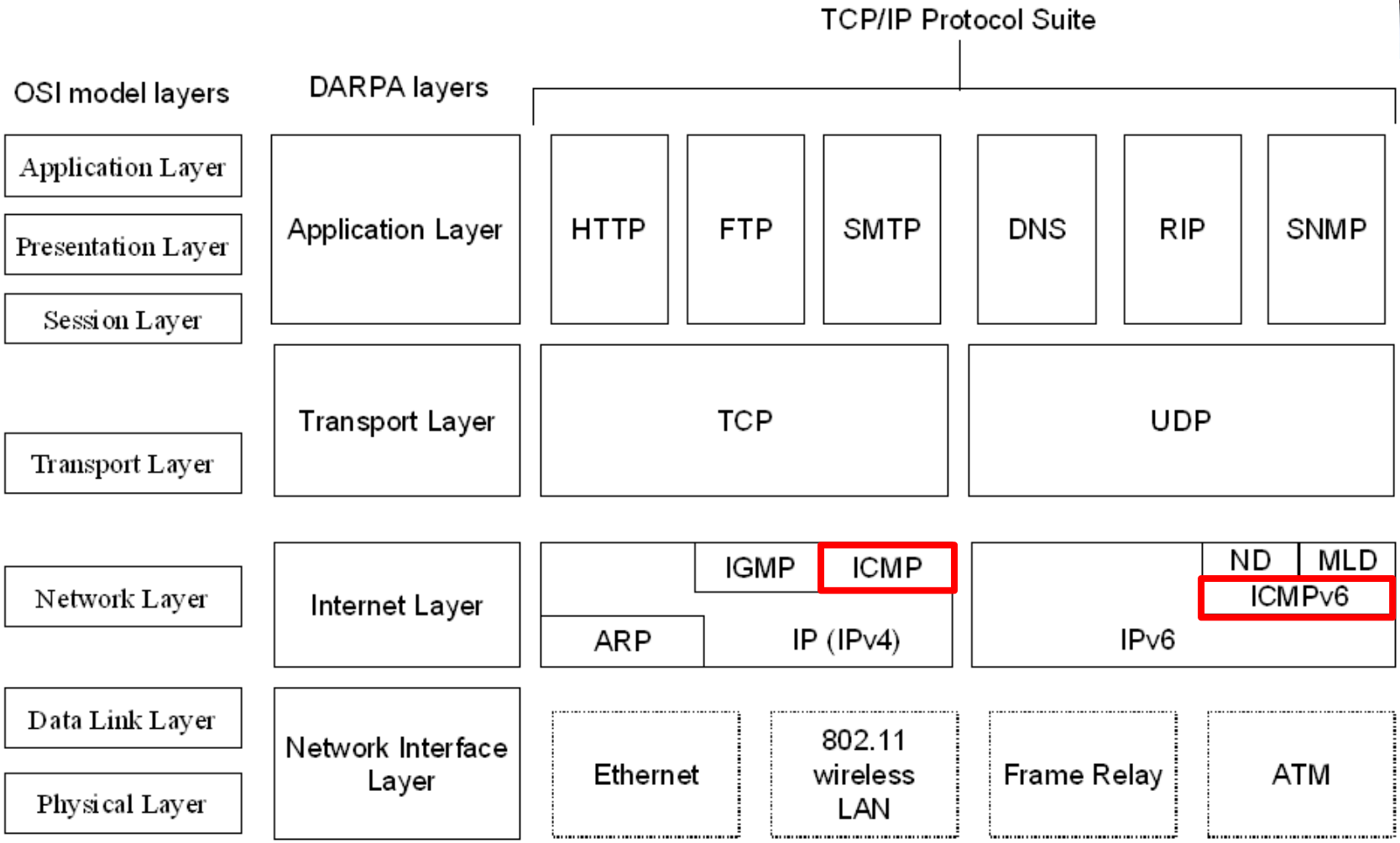
- In IPV6, 8 octets are usually allowed for the Host ID. Since the Ethernet MAC address is only 6 octets, it is recommended to be used as the Host ID in the IPV6 address.
- This means the physical address and the IP address are bound so sending packets to another station on a LAN is trivial.
- This is known as “Direct Mapping.”
- If direct mapping is not used, IPV6 provides another technique through “Neighbor Discovery Protocol” (NDP). NDP will be described later.

Internet Control Message Protocol (ICMP)

- Before we go on to some of the higher level protocols, we need to look at another Layer 3 protocol, the Internet Control Message Protocol (ICMP).
- The Internet is a virtual network and, therefore, cannot rely on the physical network to report problems. The physical networks are independent and *cannot* report problems across other networks. The IP protocol must handle problem reports.
- Some problems could be: hardware or link failures, destination machine unreachable (maybe turned off), the hop limit expires, or if a node is so overloaded it must discard datagrams.

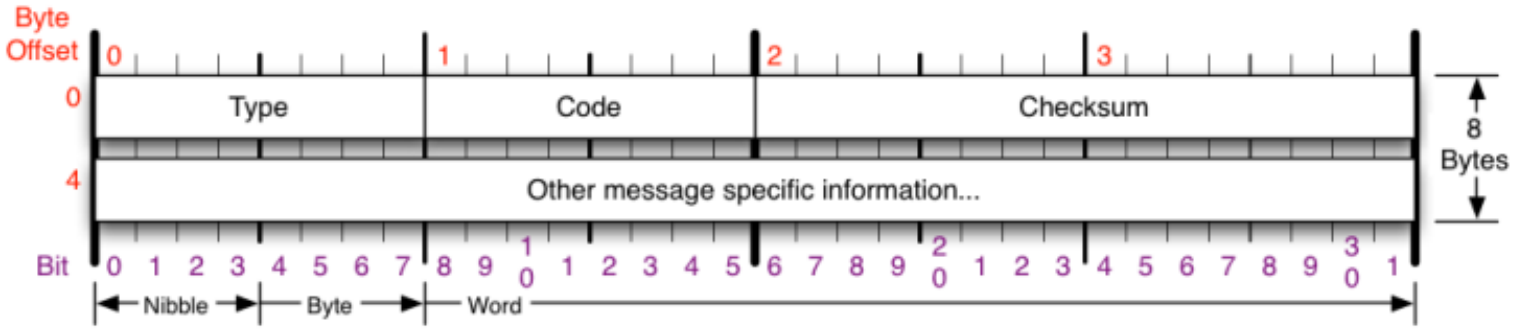
ICMP in the Protocol Layers

- While ICMP is a Layer 3 protocol, it is carried in an IP packet.



ICMP Message Format

- The exact format of the ICMP message depends on what the message is. But all ICMP messages begin with the 4 octets shown below:
- The Type and Code fields indicate what ICMP message follows. Some examples are given in the figure (for IPV4).



ICMP Message Types

Type	Code/Name
0	Echo Reply
3	Destination Unreachable
	0 Net Unreachable
	1 Host Unreachable
	2 Protocol Unreachable
	3 Port Unreachable
	4 Fragmentation required, and DF set
	5 Source Route Failed
	6 Destination Network Unknown
	7 Destination Host Unknown
	8 Source Host Isolated
	9 Network Administratively Prohibited
	10 Host Administratively Prohibited
	11 Network Unreachable for TOS
3	Destination Unreachable (continued)
	12 Host Unreachable for TOS
	13 Communication Administratively Prohibited
4	Source Quench
5	Redirect
	0 Redirect Datagram for the Network
	1 Redirect Datagram for the Host
	2 Redirect Datagram for the TOS & Network
	3 Redirect Datagram for the TOS & Host
8	Echo
9	Router Advertisement
10	Router Selection
11	Time Exceded
	0 TTL Exceeded
	1 Fragment Reassembly Time Exceeded
12	Parameter Problem
	0 Pointer Problem
	1 Missing a Required Operand
	2 Bad Length
13	Timestamp
14	Timestamp Reply
15	Information Request
16	Information Reply
17	Address Mask Request
18	Address Mask Reply
30	Traceroute

Checksum

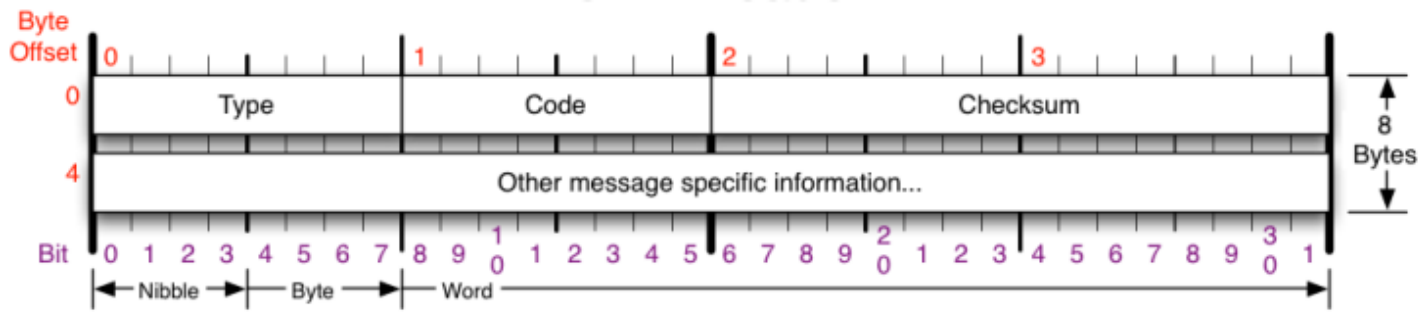
Checksum of ICMP header

RFC 792

Please refer to RFC 792 for the Internet Control Message protocol (ICMP) specification.

ICMP Message Format

- The Checksum is the 16 bit 1s complement addition and is over this header and the ICMP data. It is computed with the Checksum field set to zero and then the result of the addition is one's complemented and stored in the Checksum field. See RFC 1071.



ICMP Message Types

Type	Code/Name
0	Echo Reply
3	Destination Unreachable
	0 Net Unreachable
	1 Host Unreachable
	2 Protocol Unreachable
	3 Port Unreachable
	4 Fragmentation required, and DF set
	5 Source Route Failed
	6 Destination Network Unknown
	7 Destination Host Unknown
	8 Source Host Isolated
	9 Network Administratively Prohibited
	10 Host Administratively Prohibited
	11 Network Unreachable for TOS
3	Destination Unreachable (continued)
	12 Host Unreachable for TOS
	13 Communication Administratively Prohibited
4	Source Quench
5	Redirect
	0 Redirect Datagram for the Network
	1 Redirect Datagram for the Host
	2 Redirect Datagram for the TOS & Network
	3 Redirect Datagram for the TOS & Host
8	Echo
9	Router Advertisement
10	Router Selection
11	Time Exceded
	0 TTL Exceeded
	1 Fragment Reassembly Time Exceeded
12	Parameter Problem
	0 Pointer Problem
	1 Missing a Required Operand
	2 Bad Length
13	Timestamp
14	Timestamp Reply
15	Information Request
16	Information Reply
17	Address Mask Request
18	Address Mask Reply
30	Traceroute

Checksum

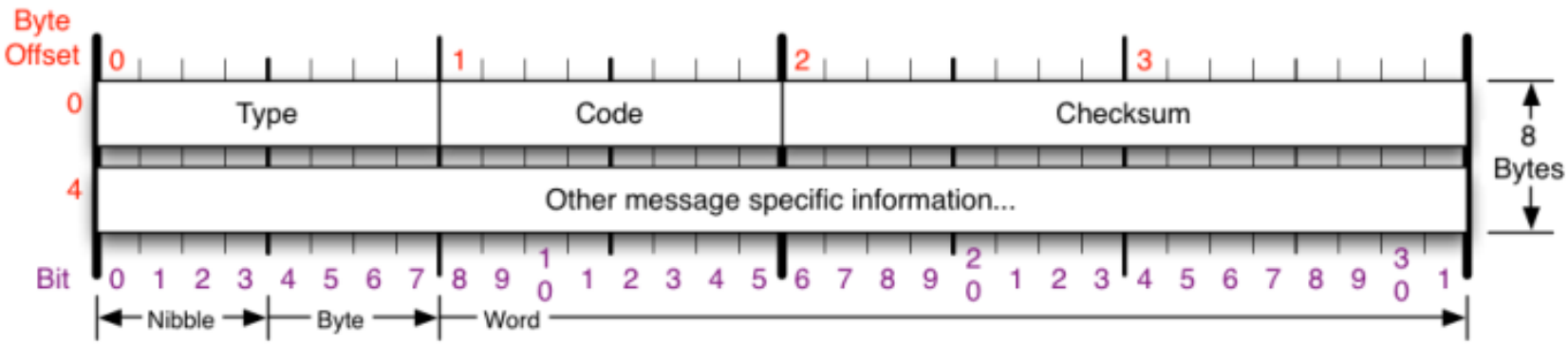
Checksum of ICMP header

RFC 792

Please refer to RFC 792 for the Internet Control Message protocol (ICMP) specification.

ICMPv6

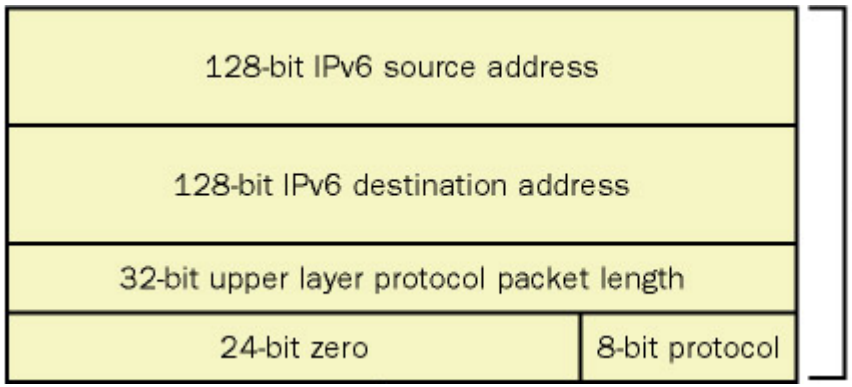
- For IPV6, the same four octets are used at the beginning of the packet, but the Type and Code fields use different numbers than IPV4 (See RFC 4443).
- The Checksum is also calculated differently. The checksum is calculated as the 1s complement addition over the header and the ICMP data, but it also includes a pseudo-header in the calculations. The pseudo-header is shown in the next slide.



ICMPv6 Checksum Calculation

- When the checksum is calculated, the following pseudo-header is included in the calculation. This header is not transmitted – the data is taken from the IP Base Header – it is only used to calculated the checksum.
- The purpose of including the pseudo-header is to make sure the ICMP message returns to the correct sender, since the source and destination addresses are included in the pseudo-header.
- This also provides some security from spoofed messages.

The packet length is the length of the ICMPv6 packet, not including this pseudo-header.



The protocol field is 58 for ICMPv6

Autoconfiguration (DHCP and NDP)

- We're finished with Layer 3 (I'm not going to cover IGMP) but before we start on the higher layers, I want to discuss how a station gets it's IP address.
- In IPV4, this is generally done with Dynamic Host Configuration Protocol (DHCP) and we'll start with that.
- DHCP uses the User Datagram Protocol (UDP) which we haven't covered yet, so you'll have to bear with me on that.
- After we finish with DHCP, we'll look at how it's done in IPV6.

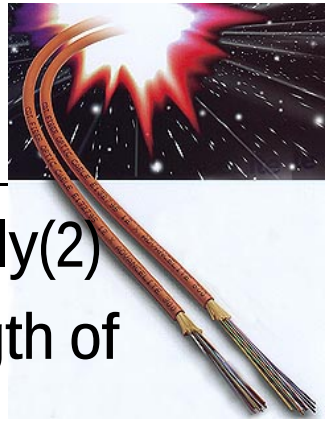
DHCP (IPV4)

- When a station comes up on a LAN (for example), it needs to obtain an IP address, the address mask, the address of the gateway router, and the address of a name server (DNS – to be discussed later).
- All of this information can be provided by a special server, but how can the station contact this server without an IP address?
- The answer is by using the broadcast IP address of 255.255.255.255 with port 67 (ports will be discussed later). I'll describe the operation on an Ethernet LAN.

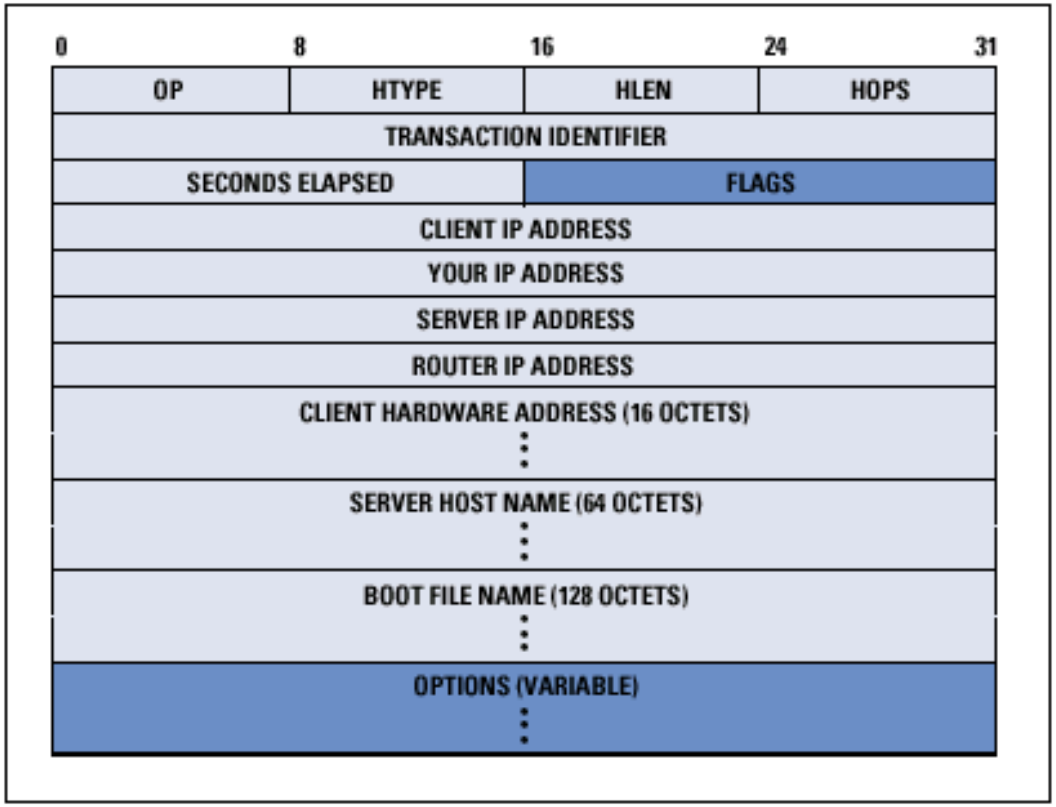
DHCP (IPV4)

- At Layer 2, the broadcast IP address resolves to an Ethernet broadcast address. All stations (including routers and servers) on the LAN will receive and process the message, but only the DHCP server will respond.
- The server will respond with the needed information via a broadcast datagram because at this point, the station still does not have an IP address.
- Let's look at the DHCP packet. The request sent by the client is called the DHCPDiscover and the reply sent by the server is called the DHCPOffer.

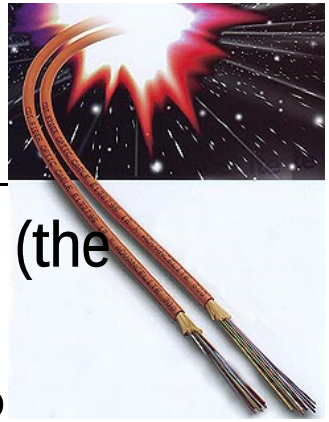
DHCP Packet (IPV4)



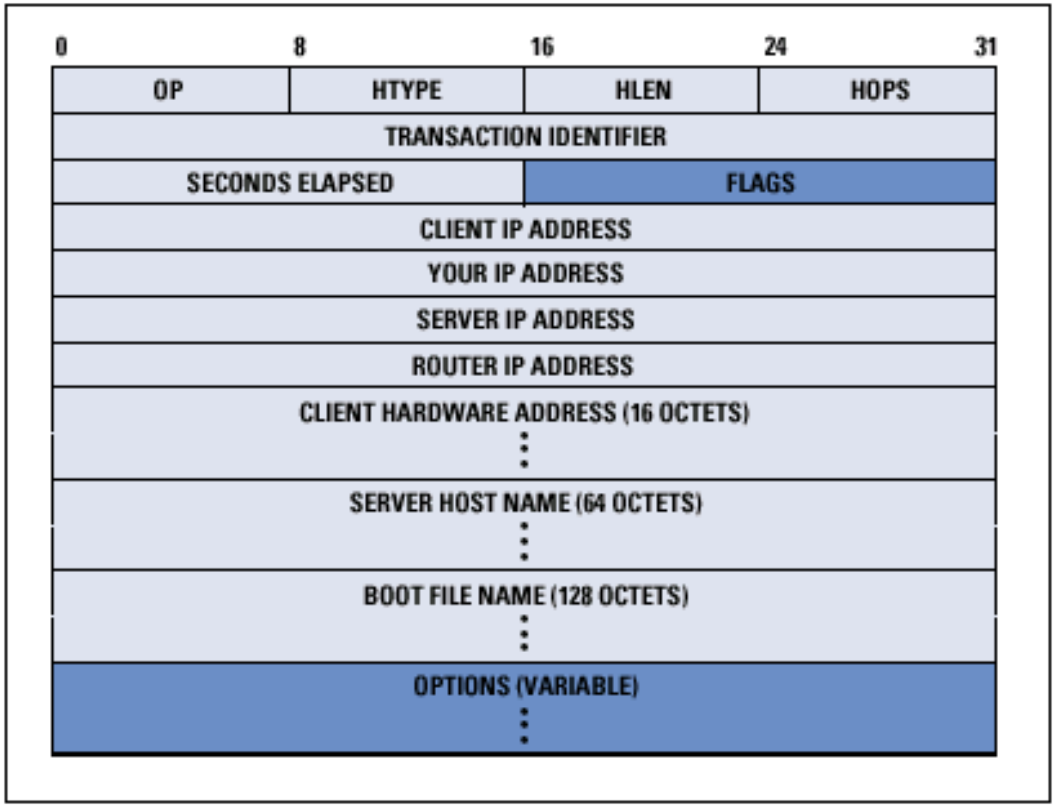
- OP specifies whether the message is a request (1) or reply(2)
- HTYPE and HLEN specify the hardware type and the length of the hardware address.
- The client puts 0 in the HOPS field and DHCP server increments the count.
- Transaction ID is a number to match response to request.
- Seconds is how long since the client started to boot.
- Only the first bit is used in Flags, and it indicates whether the response should be by broadcast.



DHCP Packet (IPV4)

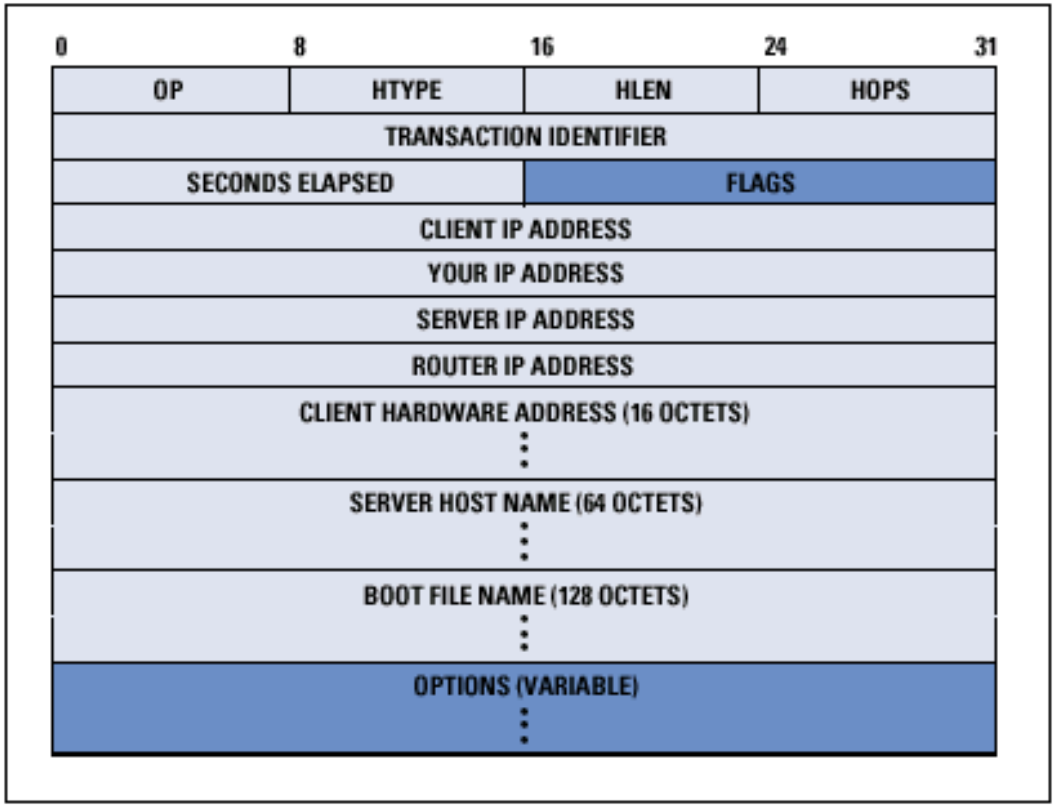


- Client IP Address is the IP address of the client, if known (the client may have an IP address but need other info).
- Your IP Address is where the server puts the assigned IP address.
- Server IP Address and/or Server Host Name are used by the client if it wants a response from a specific DHCP server.
- Router IP Address will normally be set to 0 by the client.
- I won't cover the Boot File Name field.



DHCP Packet (IPV4)

- The client will usually include certain information in the Options field, specifically Option 53 (DHCP discover), and Option 55 with a list of information requested: subnet mask (1), router (3), Domain Name (15), and Domain Name server (6).
- The server's response will have Your IP Address, Server IP Address (it's IP address), Client H/W Address (MAC address), with the rest of the data in the Options field.
  1 = Subnet mask
  3 = Gateway router
  6 = DNS servers
  and more.



DHCP Packet (IPV4)

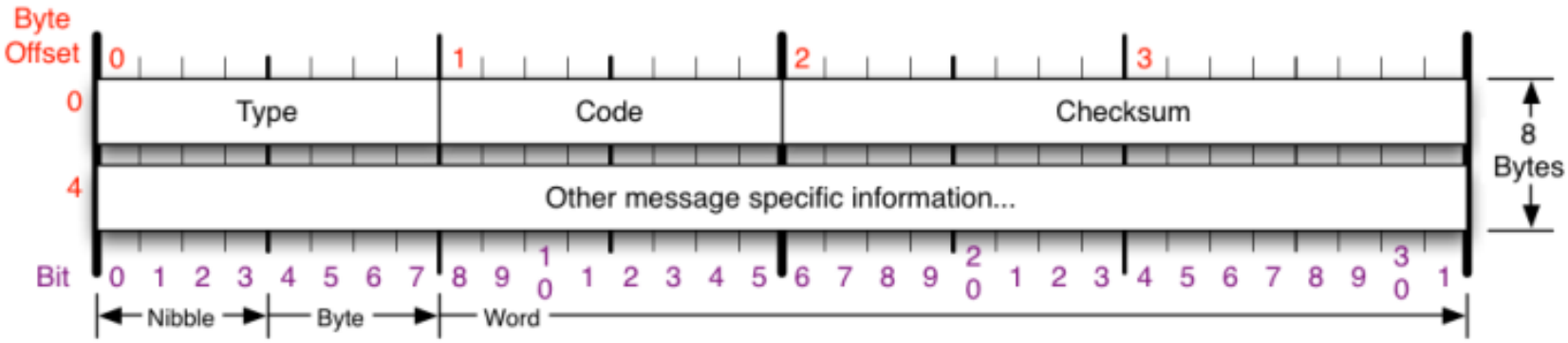
- Since IP is not a reliable protocol, the client must inform the DHCP server that it has received the DHCPOffer and has accepted the data.
- It sends a DHCPRequest packet via broadcast with the information that was in the DHCPOffer. The reason for the broadcast is that the client may receive DHCPOffers from multiple servers. The broadcast tells all the servers which offer has been accepted.
- The server whose offer has been accepted sends a DHCPAck via the IP address back to the client to finish the sequence.

IPV6

- There are two ways for a station to obtain the necessary information at startup – through DHCPv6 (RFC 3315) and through Neighbor Discovery Protocol (NDP – RFC 4861).
- The techniques are called, respectively, managed and unmanaged. In some papers, it's called "stateful" and "stateless"
- DHCPv6 is used to assign IP addresses to stations, and the Host ID portion of the address may not be the station's MAC address.
- DHCPv6 works essentially the same as IPV4 DHCP, but it's more complex in the details. Since this is a survey class, I won't go into those details. You know conceptually how it has to work.

IPV6 NDP

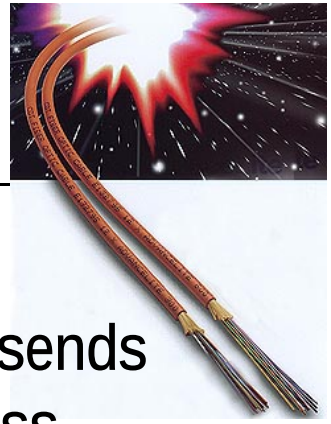
- IPV6 introduced a new, unmanaged or stateless, way for stations to learn the information required to participate in the Internet. The advantage of the unmanaged approach is that a DHCP server is not required.
- NDP uses the ICMPv6 four octet header format. The Type field is used to indicate the function of the message.
- There are five message types: Router Solicitation (133), Router Advertisement (134), Neighbor Solicitation (135), Neighbor Advertisement (136), and Redirect (137).



IPV6 NDP

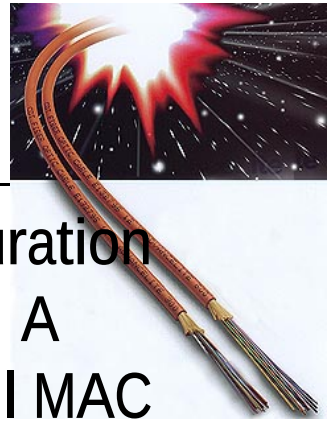
- The Router Solicitation message is sent to prompt routers to respond.
- The Router Advertisement message is sent periodically or in response to a Router Solicitation message and gives information about it's ability to handle off-link traffic.
- The Neighbor Solicitation message is sent to obtain the MAC address of a neighbor or to verify the neighbor is still reachable. It is also used to find certain other needed information, such as the DNS server.
- The Neighbor Advertisement message is sent in response to a Neighbor Solicitation message and contains information about the unit, including the MAC address.
- The Redirect message is used to ask a host to change it's first hop for a specific destination.

IPV6 NDP



- To find the routers available to the station, the station sends an ICMPv6 (NDP) type 133 message to multicast address FF02::2 /8, which is the “all routers multicast” address.
- To find the DNS, the station sends an ICMPv6 type 135 message to multicast address FF02::FB /8 which is the “DNS server multicast” address.
- To find the DHCPv6 server, the station would send an ICMPv6 type 135 message to multicast address FF05::1:3 /8

IPV6 Autoconfigure



- When using the unmanaged technique of auto configuration in IPV6, the station will begin by generating a Host ID. A safe Host ID would be the unit's MAC address since all MAC addresses are unique. But it could generate a random number.
- It will use the Link-Local Unicast prefix, FE80::/10, as its Network ID.
- It will make a Router Solicitation, using the multicast address FF02::2 /8 to obtain the Network ID and the address mask, as well as the IP and MAC address of the gateway router(s).
- It can then issue a Neighbor Solicitation for its IP address (if randomly generated), to make sure its address is unique. If there's no response, its address is not duplicated.

IPV6 Autoconfigure

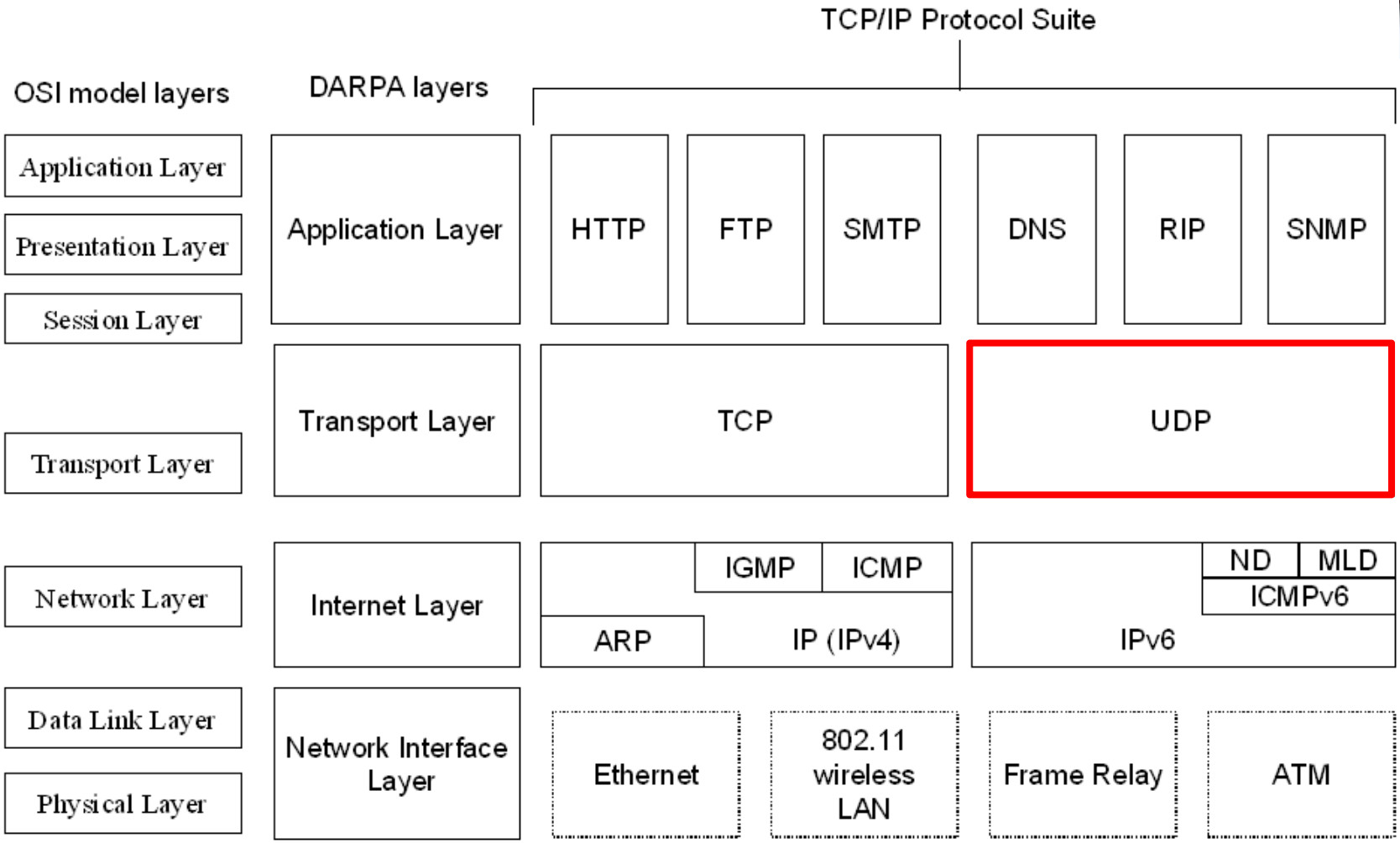
- It can then issue a Neighbor Solicitation for the DNSv6 server, using the multicast address FF02::FB /8 to obtain the IP and MAC address of the name server.

User Datagram Protocol (UDP)

- UDP provides an unreliable, best efforts, connectionless delivery service.
- Humm, haven't we heard that description before? Oh, yes, that's exactly the description of IP.
- So why do we need UDP? Why not just put the data into the IP packets?
- Because UDP introduces the concept of "ports", a very important part of the Internet suite.

UDP in the Protocol Layers

- UDP is a Layer 4 protocol, carried in an IP packet.



Internet Ports

- When we want to communicate with an application on another host (a server, actually), how can we direct our data to the correct application?
- We do it by using a “well known” port number which has been assigned to that application.
- For example, the File Transfer Protocol (FTP) application uses ports 20 and 21. When one station wants to connect to the FTP application on another host, it will use the ports 20 and 21.
- When those packets arrive at the server, the layer 4 protocol knows that packets with those ports go to the FTP server application.

Internet Ports

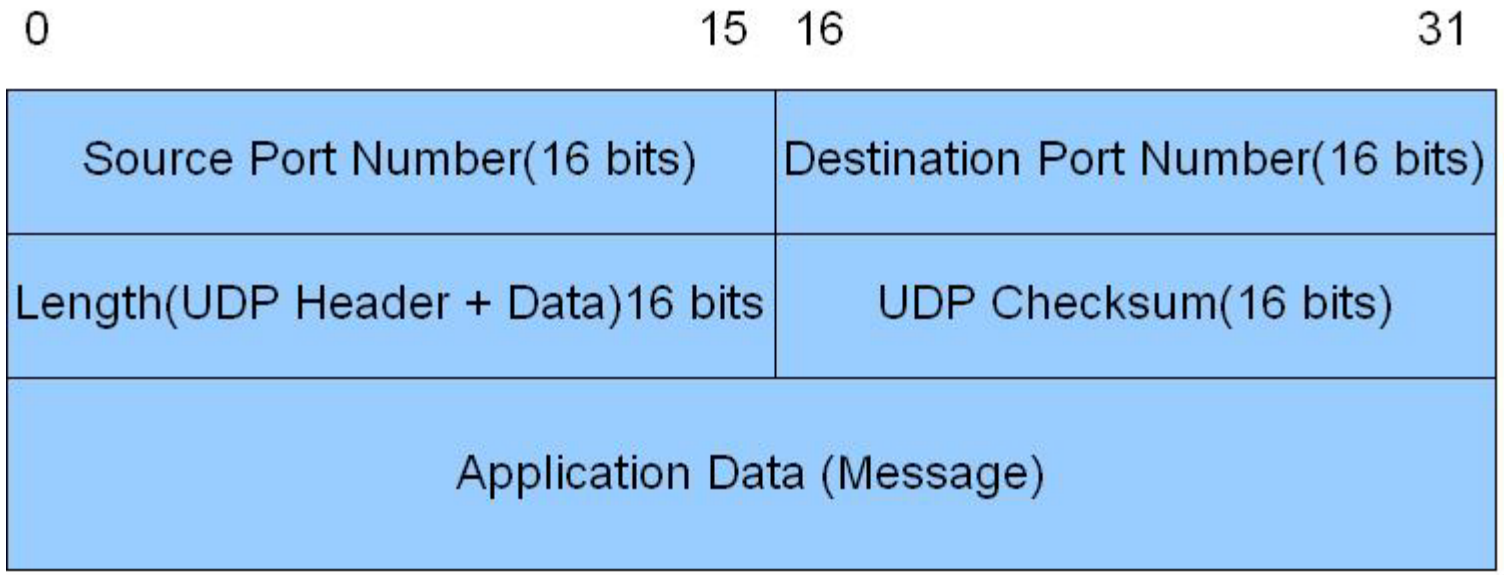
- That answers the question of how data gets to an application on a server, but how does the application send data back to the originating host?
- When the originating host initiated the sequence for communicating with the server, it asked the operating system for a port number. The OS assigned a port number and also allocated space for buffers.
- The UDP packets from the originating host includes its port number as well as the destination port number. So the server application knows how to send data back to the originator.

Internet Ports

- Most of the common applications used in the Internet have ports assigned to them.
- These are known as “Well Known Ports”. When a packet with one of those ports arrives, it will be sent to the registered application. You can find the list of well known ports in many places on the Internet.
- Any other application can choose from the list of non-registered ports and can use that port for communication with an equivalent application on another computer.

UDP Message Format and Header

- UDP takes the data from the application and adds an eight octet header.
- The source and destination port numbers are each 16 bits, so port numbers can go from 0 to 65,535.
- The Length field is the length of the header plus the data.

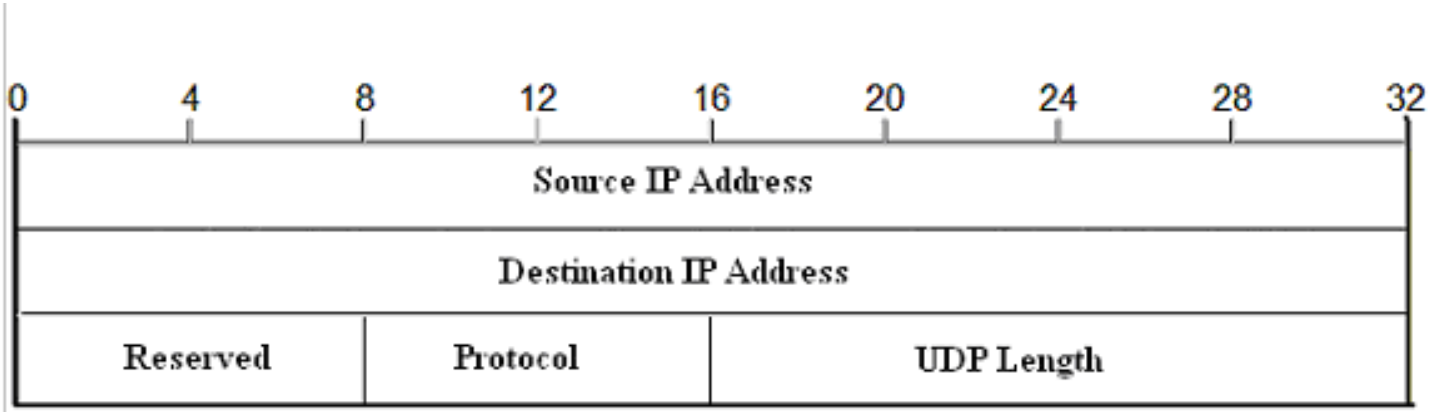


UDP Checksum – IPV4

- The UDP Checksum takes more explanation.
- In IPV4 it is optional. If the field is zero, it's an indication that the checksum has not been computed and should be ignored.
- The IPV4 checksum is computed over the header and the data. Remember that IP only does a checksum over the header so this checksum is needed to verify the data.
- In 1s complement arithmetic, there are two ways to represent zero – as all zeroes and as all ones. Should the checksum compute to all zeroes, it is complemented and sent as the all 1s version of zero.

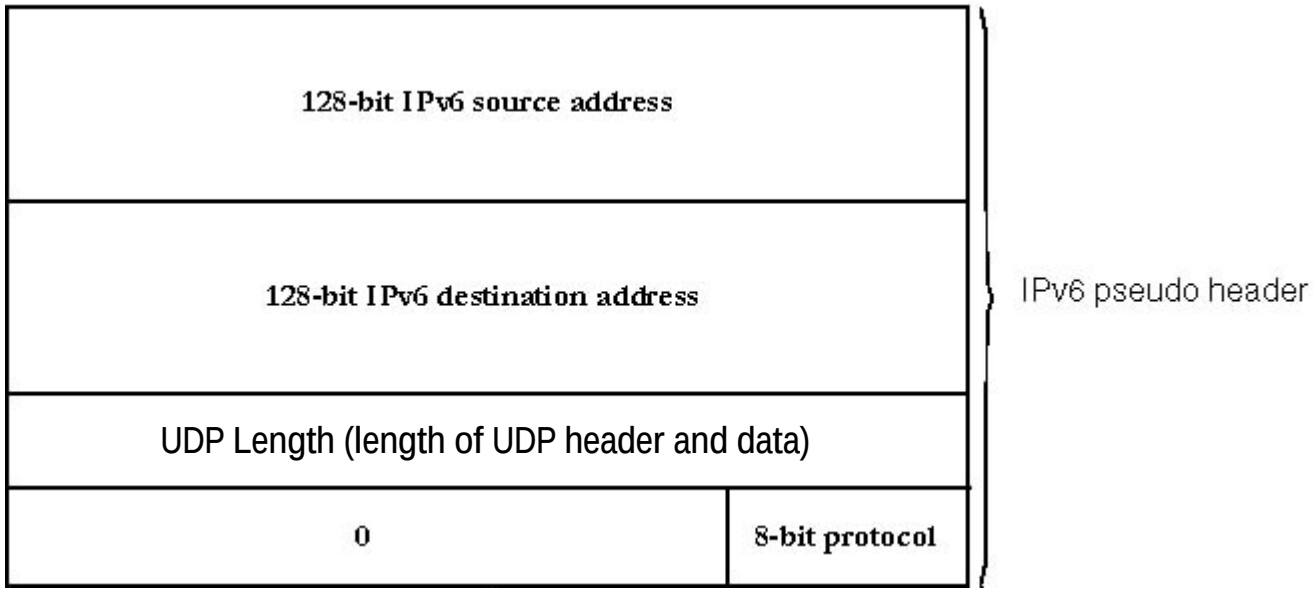
UDP Checksum – IPV4

- In IPV4, the checksum includes a pseudo header in the checksum calculation.
- The pseudo header is 12 octets and contains the fields shown below – the data in the fields is taken primarily from the IP header, except for the Length field, which is the length from the UDP header. The Reserved field is all zeroes.



UDP Checksum – IPV6

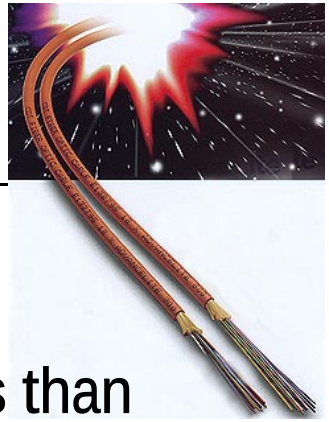
- In IPV6, the UDP checksum is required.
- A pseudo header is included in the checksum but the pseudo header is different from the one used in IPV4.
- The IPV6 pseudo header is 40 octets and the information is primarily taken from the IP header.
- The protocol is 17 (UDP) and the length is taken from the UDP header.



UDP Checksum

- Just like the IP checksum, the UDP checksum is the complement of the 1s complement addition, taken 16 bits at a time.
- You may ask “Why do they complement after doing the 1s complement addition?” The reason is that when the packet is checked at the receiver, 1s complement addition which includes the checksum will always give an all 1s result. This makes it easy to verify that the packet had no bit errors.
- The UDP checksum is over the pseudo header (40 octets), the UDP header (6 octets without the checksum), and the data (can be a lot of octets). This large coverage reduces the detection accuracy of the checksum. A CRC would be better.

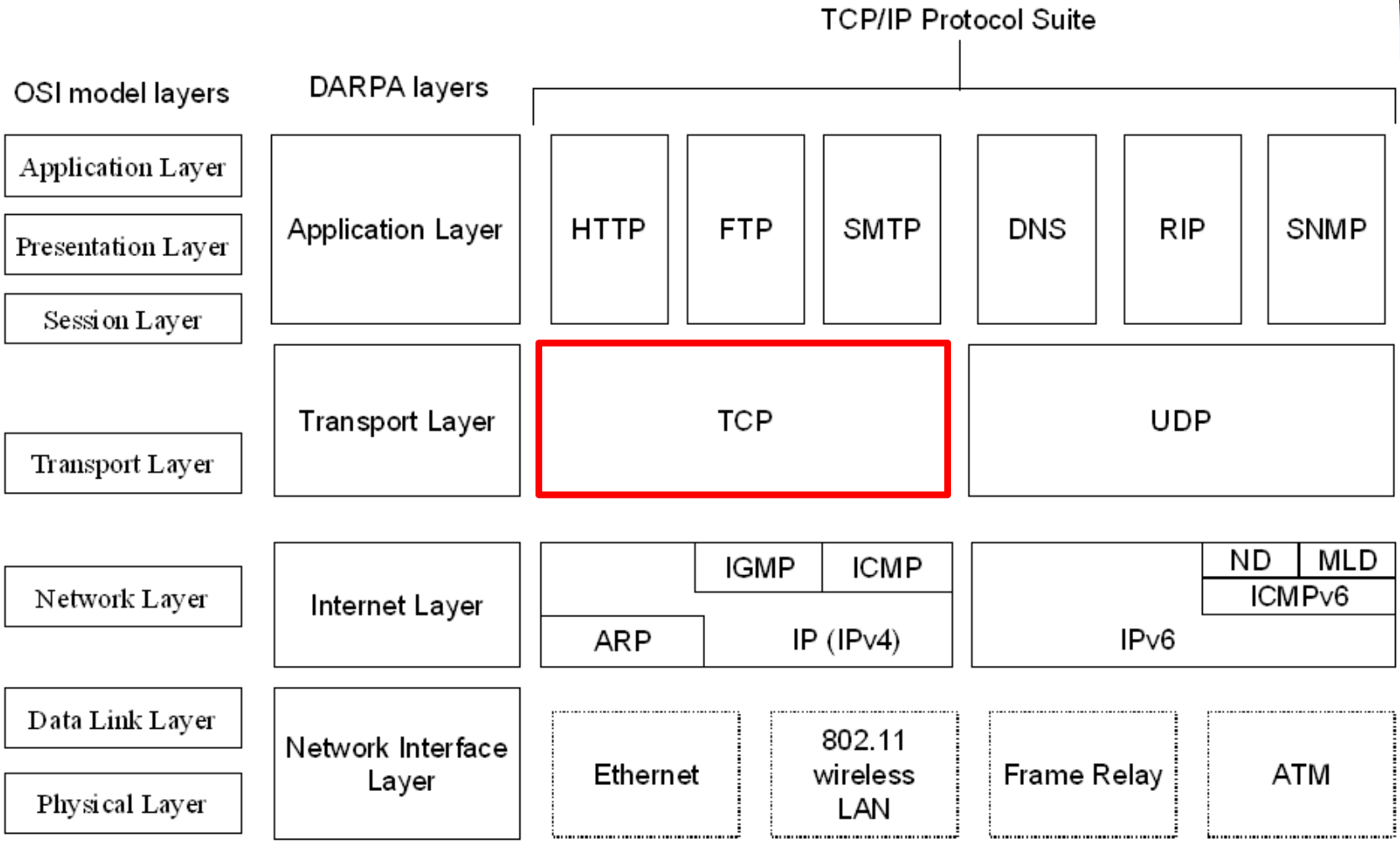
Transmission Control Protocol (TCP)



- Many applications need more reliable communications than is provided with UDP.
- That reliability is provided by the Transmission Control Protocol (TCP), and that reliability makes TCP a more complex protocol than UDP.

TCP in the Protocol Layers

- TCP is a Layer 4 protocol, carried in an IP packet.



Characteristics of TCP

- TCP has the following characteristics: (1) Stream Orientation, (2) Virtual Circuit Connection, (3) Buffered Transfer, (4) Unstructured Stream, and (5) Full Duplex Communication.
- Steam Oriented – This means that a TCP connection is abstracted as simply as continuous stream of octets. The data is not viewed as being sent in "packets" or any other abstraction. This means that TCP must keep track of which octets have been sent and received on an octet basis. It also means that when a retransmission occurs, the retransmission may not be of the same number of octets as the first transmission (usually more).

Characteristics of TCP

- Virtual Circuit Connection – Before data can be communicated, the two end stations must agree to establish a TCP connection. Usually, one station initiates the connection, such as when you "go" to a web site. The two stations must agree to establish the connection, agree on the details of the connection, and inform their respective application programs that the connection is established.
  The term "virtual circuit" is used for this connection because it can be visualized as similar to a voice telephone call, except that the data is carried in packets.
  At the time TCP was defined, the circuit paradigm was the way people thought of communications. The concept of "packets" was a new idea.

Characteristics of TCP

- Buffered Transfer – TCP receives data from an application and buffers it in memory prior to packetizing it for transfer. The way TCP segments the data for transfer is completely independent of the application. The application simply provides a stream of data to TCP.
  This can cause problems. For example, if the data is from a TTY type terminal that the user is typing on, the characters must be sent, received, processes, and echoed before they appear on the TTY (paper or screen). To deal with this problem, TCP implements a “push” mechanism to cause data to be sent immediately.

Characteristics of TCP

- Unstructured Stream – TCP does not, and can not, know what type of data is being sent. It's just a stream of octets to TCP. If the data has some structure, for example, personnel records, the application program must be able to extract the records from the stream of octets.

Characteristics of TCP

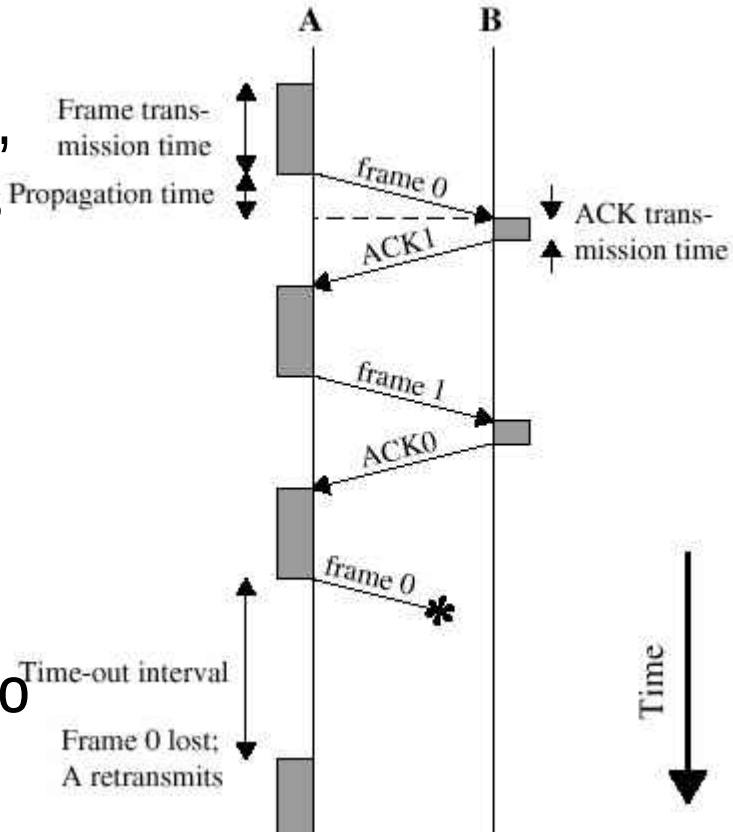
- Full Duplex Communication – TCP Virtual Circuits are all full duplex. Conceptually, in TCP they are two independent data streams, with no interaction between them.
  That's not completely true, however, because each stream carries information about the other stream, specifically acknowledgments of data received in the other stream,

How Does TCP Provide Reliability?

- Since the protocol layers underlying TCP only provide “best efforts” delivery, how does TCP provide reliability?
- TCP requires the receiver to acknowledge the data sent, and if it is not acknowledged, the sender will resend it.
- While that statement is simple, the implementation is not. Let’s look deeper into how it’s done.

Simple "Send and Wait"

- One method of acknowledgement (ACK) is for the sender to wait for an ACK after each packet.
- The sequence is shown in the figure below.
- The packet is sent to the receiver, who examines the packet for errors, and if no errors are detected, sends an ACK back to the sender.
- The sender starts a clock when the packet is sent and if no ACK is received in that time, resends the packet.
- This may mean the receiver gets two copies, but just throws one away.

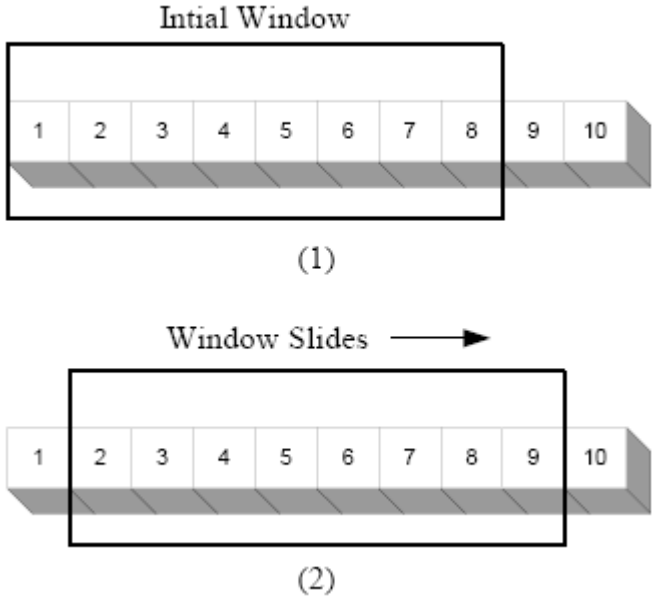


"Send and Wait"

- Acknowledgment systems must include a way for the receiver to detect duplicate transmissions. This is generally done by some type of sequence number. In TCP, the octets to be transmitted are counted and the sequence number sent with the packet. The receiver uses this sequence number to acknowledge data received.
- Any type of ACK system requires the sender to keep a copy of any data sent until it is ACK'ed.
- The major problem with "send and wait" is that the network is not used very efficiently. There's a lot of "dead" time in the communication.

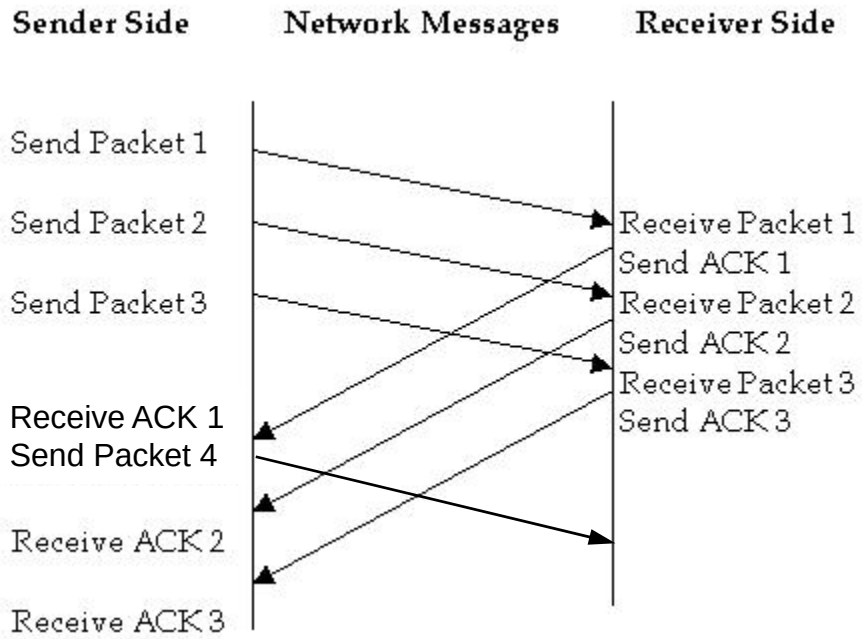
"Sliding Window"

- A method which attempts to address the transmission efficiency problem is the concept of a "Sliding Window".
- A window is the number of packets (not octets) that the sender is permitted to send without receiving an ACK for those packets. That is, it is the number of packets that the sender can have waiting for ACKS.
- For example, see the figure to the right. The window size is 8, so the sender will send 8 packets and then stop, waiting for an ACK.
- When it receives and ACK for the first packet, the window logically slides right and another packet can be sent.



Sliding Window

- Here's a network diagram for a window size of 3.
- You can see that three packets are sent before ever receiving an ACK.
- After receiving an ACK for packet 1, the sender would immediately send packet 4, etc.

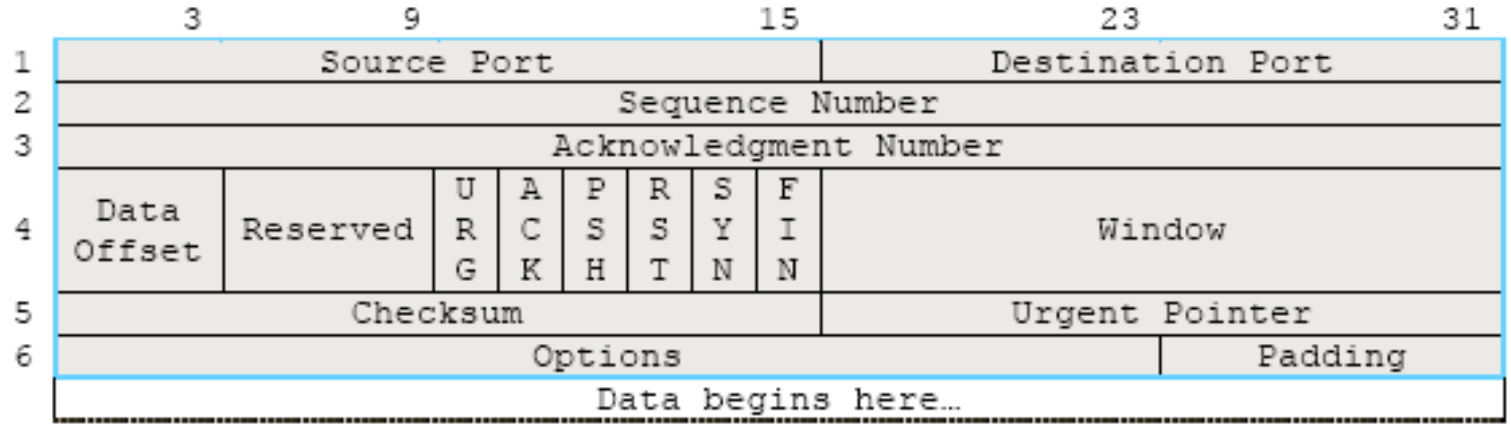


TCP Connection IDs

- Earlier, we discussed UDP ports and how UDP uses the port number to connect to applications that use “well known ports”. Return data is sent to the port specified by the initiator.
- TCP uses a slightly different scheme, a “connection” abstraction. A connection is identified by the IP address and port number of the initiator AND the IP address and port number of the server application.
- Perhaps the reason for including the IP address and port addresses of the two ends is that TCP is a full duplex protocol. So the connection identification provides complete information for two way communications.

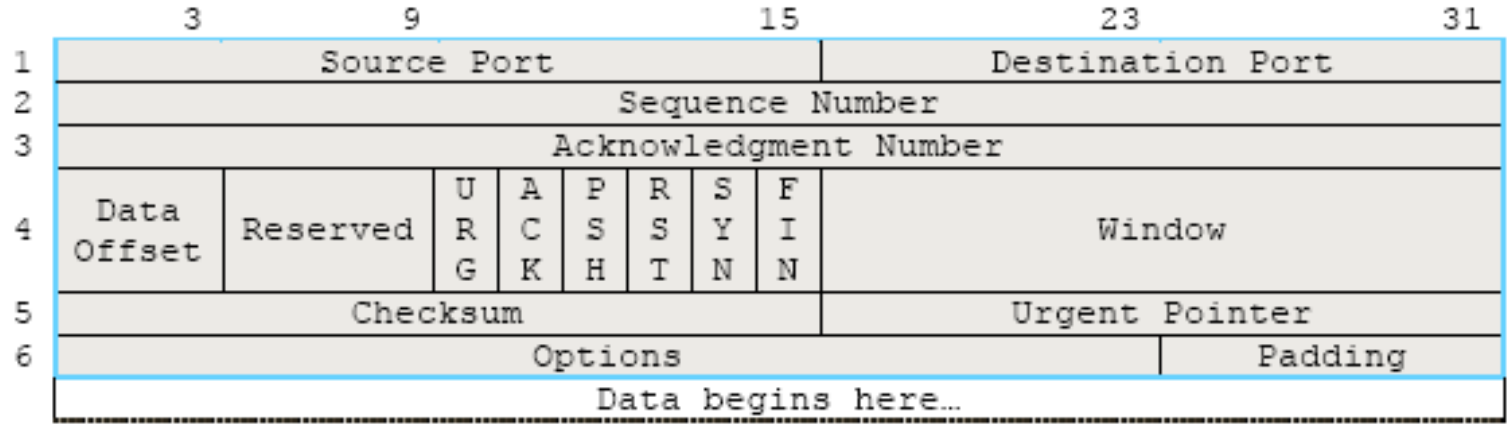
TCP Header

- The Source Port and Destination Port fields contain the port numbers of source and destination.
- The Sequence Number identifies the octet location of the start of the data contained in this packet in the data stream.
- The Acknowledgement Number is a bit difficult to explain. Let's say that the last octet of the last packet acknowledged was 100, and the next packet had 10 octets in it. This field would have the value 111.



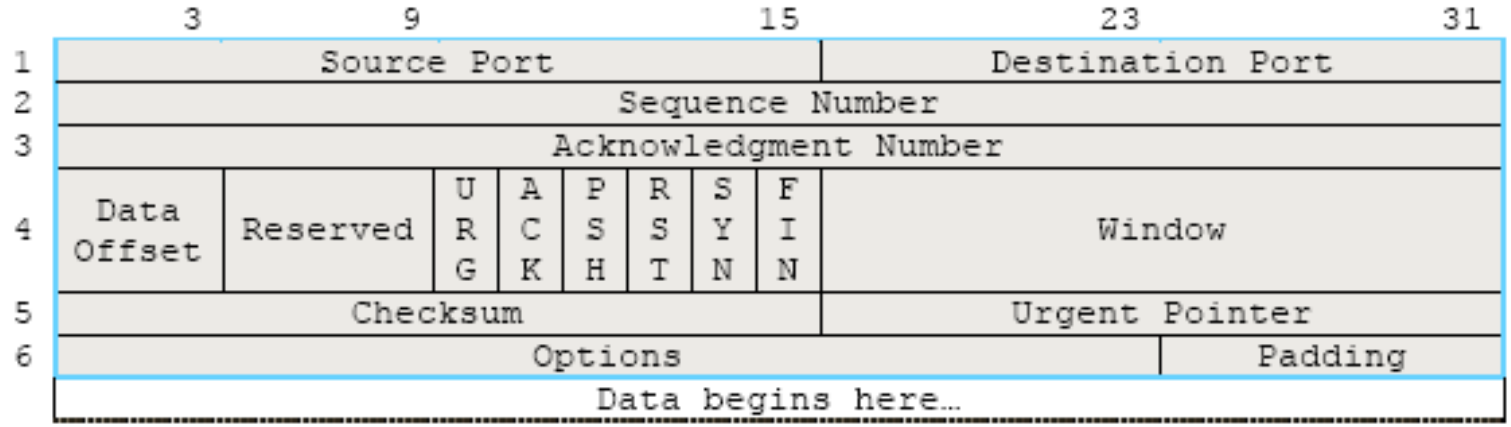
TCP Header

- The Data Offset is the length of the header in 32 bit units. It indicates the location of the start of the data. This is needed because there can be options in the header.
- The Reserved field is not used.
- The URG bit, if set, indicates that the Urgent Pointer is valid.
- The ACK bit, if set, indicates that the Acknowledgment Number field is valid (this packet ACKs data).



TCP Header

- The PSH bit, if set, indicates that this segment is pushed.
- The RST bit, if set, resets the connection.
- The SYN and FIN bits are used when initiating or terminating a connection.
- The Window field indicates how many octets the sender can accept in its buffers.



TCP Header

- The Checksum is the complement of the 16 bit 1S complement addition of the pseudo-header, the TCP header, and the data.
- For operation on IPV4, the checksum is calculated over the fields shown below.

TCP pseudo-header for checksum computation (IPv4)

Bit offset	0–3	4–7	8–15	16–31
0	Source address			
32	Destination address			
64	Zeros		Protocol	TCP length
96	Source port			Destination port
128	Sequence number			
160	Acknowledgement number			
192	Data offset	Reserved	Flags	Window
224	Checksum			Urgent pointer
256	Options (optional)			
256/288+	Data			

TCP Header

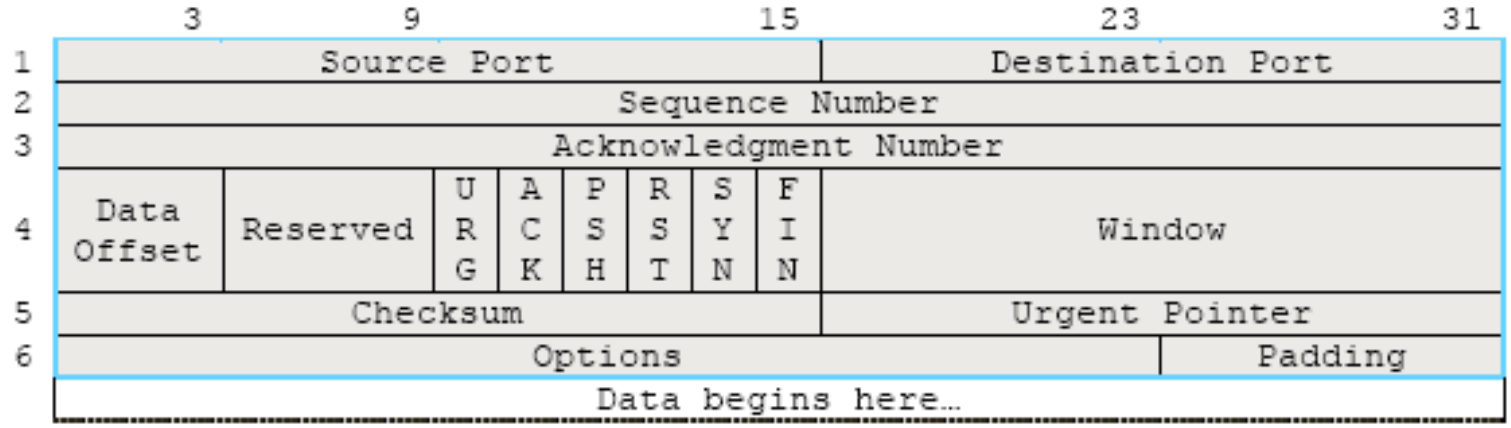
- For IPV6, the checksum is calculated over the fields shown t below.

TCP pseudo-header for checksum computation (IPv6)

Bit offset	0–7	8–15	16–23	24–31
0	Source address			
32				
64				
96				
128	Destination address			
160				
192				
224				
256	TCP length			
288	Zeros			Next header
320	Source port		Destination port	
352	Sequence number			
384	Acknowledgement number			
416	Data offset	Reserved	Flags	Window
448	Checksum			Urgent pointer
480	Options (optional)			
480/512+	Data			

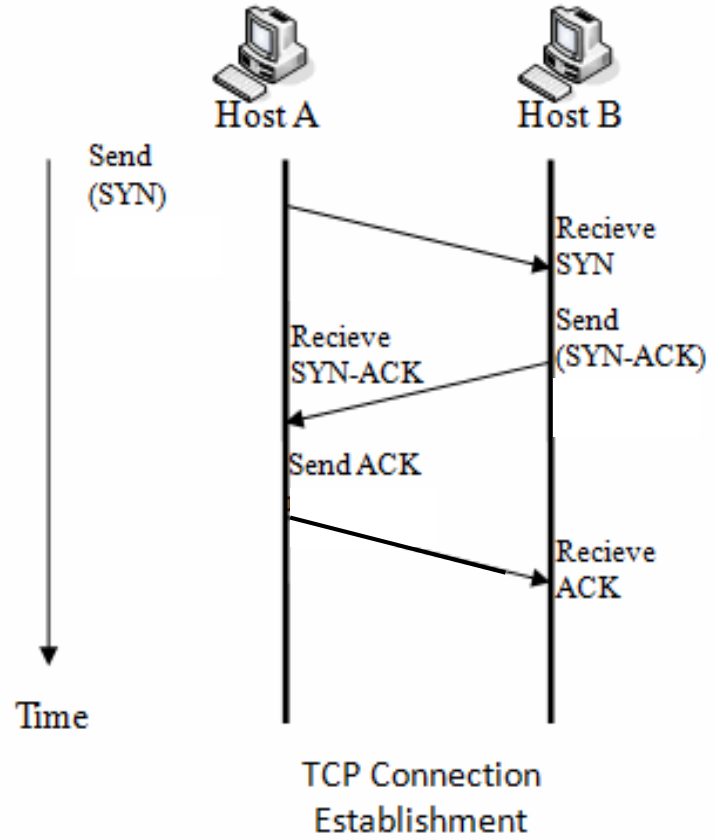
TCP Header

- The Urgent Pointer is only valid if the URG bit is set. If set, this field is an offset from the Sequence Number to the last octet of urgent data.
- There are a number of Options, and those will not be described here.



Establishing a TCP Connection

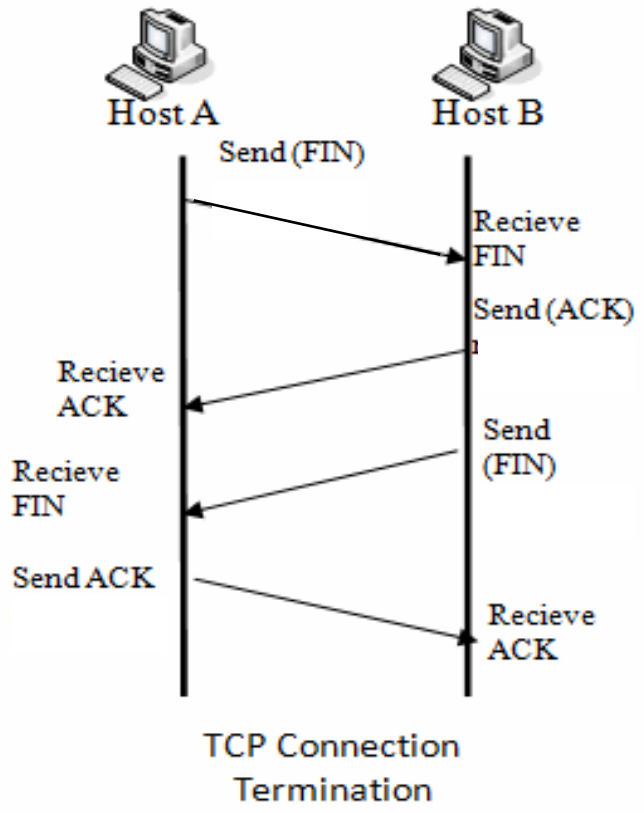
- TCP uses a three-way handshake to establish a connection.
- The three messages allow each side to know that the other side has agreed to the connection.
- The initiator (Host A in the diagram) sends a SYN message with its initial sequence number, x.
- Host B responds with its initial sequence number, y, and ACKs Host A's number by sending x+1.
- Host A ACKs Host Bs initial sequence number by sending y+1.
- The two hosts can now exchange data.



TCP Connection Establishment

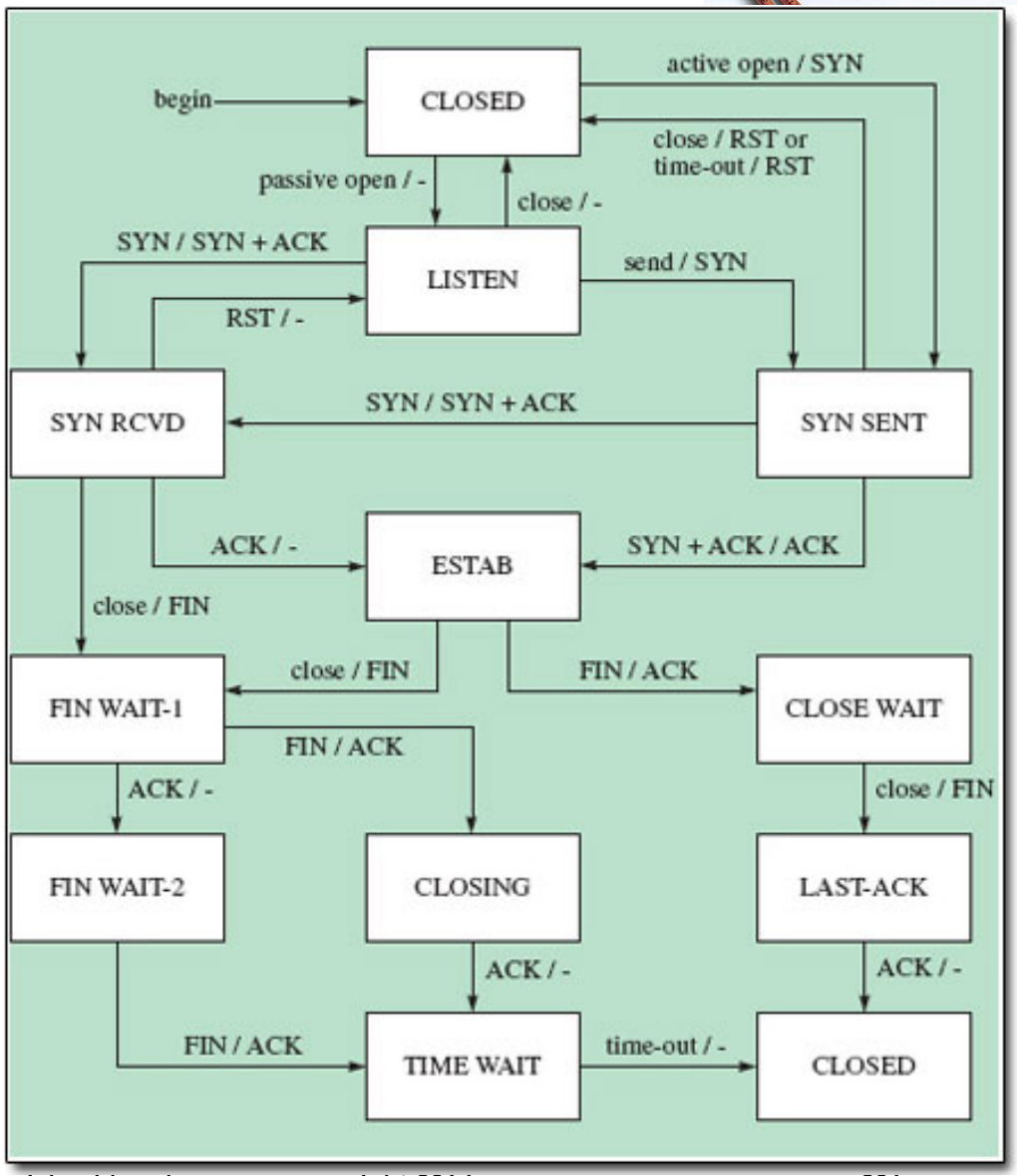
Closing a TCP Connection

- To close a connection, a three-way handshake is used.
- When a host (Host A in this diagram) is finished sending data, it will send a packet to Host B with the FIN bit set.
- Host B will inform the application that there's no more data, and send an ACK to Host A.
- When Host B is finished sending data, it will send a packet to Host A with the FIN bit set.
- Host A will inform the application of the end of stream and send an ACK to Host B.
- The two hosts will delete all resources allocated to the connection.



TCP State Machine

- A state machine shows the various states of the protocol with lines with arrows indicating transitions between states.
- We did not cover all the possible TCP states in this presentation.



Forwarding

- For many people, the Internet is a mystery. How do those packets find their way through the giant complex maze that's the Internet?
- The answer is forwarding (also called "routing") and that's what we'll look at next.
- In general, the Internet consists of hosts and nodes called routers.
  - A host may make a request to another host (often called a server) for data and the server will reply with that data.
  - A router sits inside the Internet and manages the flow of packets between hosts, forwarding those packets towards the proper destination.

General Discussion of Forwarding

- The Internet is made up of interconnected physical networks, such as Ethernet LANs.
- Earlier, in the ARP section, we talked about how a host contacts another station on the same physical network by mapping an IP address to a physical address.
- But when we have to traverse multiple physical networks to reach a destination, we have to do it through the IP address, and the nodes (called routers) that interconnect those physical networks.
- Note that at the destination network, the last router in the path will map the destination IP address to a physical address to reach the destination host.

Forwarding Table

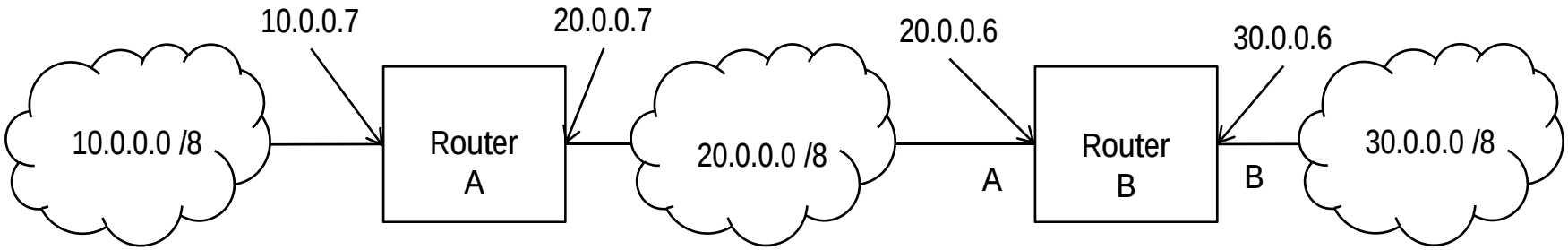
- Each router builds a forwarding table from information it receives from other routers (we'll discuss how that's done later).
- The forwarding table contains the following for each entry:
  - The IP address of the destination for this address.
  - The address mask of that entry.
  - The IP address of the next hop router.
  - The network interface to use when sending.
- The entries are put in the table with the longest mask first. So an entry with a /28 mask will go before an entry with a /16 mask.
- Let's look at the forwarding table for Router B in the diagram on the next slide.

Forwarding Table

- When Router B receives an IP packet, it will extract the IP address, and "and" the address with the mask (8 bits of 1s here) of the first entry in the table to obtain the network prefix.

To reach addresses w/ this IP	Forward to this IP	Use this port
10.0.0.0 /8	20.0.0.7	A
20.0.0.0 /8	Deliver directly	A
30.0.0.0 /8	Deliver directly	B

- If match, forward as indicated in table.
- If not, go to next entry until match made.



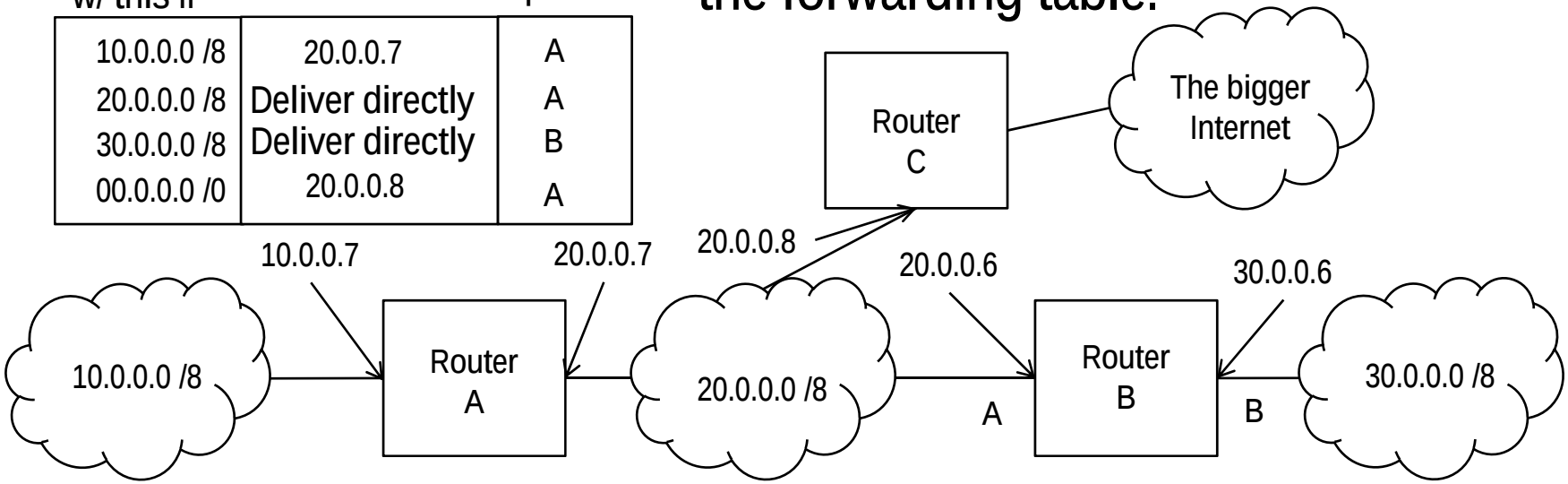
Forwarding Table

- Note that while there's an IP address to deliver the packet to when the packet is for the 10.0.0.0 network, the router will only use that IP address to obtain the physical address of the next router (Router A). It'll then put the original IP packet in a layer 2 frame and send it to that physical address. It will not put that IP address in the IP packet.
- I've conveniently left something out of this example. Suppose the packet is for an address that is not part of the 10.0.0.0, the 20.0.0.0 or the 30.0.0.0 networks. What is the router to do then?

Forwarding Table, Default Router

- A router in a corporate network, for example, should not have to keep a forwarding table that includes the entire Internet.
- To avoid this, there is the concept of a "default router" where an IP packet is forwarded if its IP address does not match any entries in the forwarding table.

To reach addresses w/ this IP	Forward to this IP	Use this port
10.0.0.0 /8	20.0.0.7	A
20.0.0.0 /8	Deliver directly	A
30.0.0.0 /8	Deliver directly	B
00.0.0.0 /0	20.0.0.8	A



Forwarding Table, Default Router

- When the destination IP address is “and” with a zero mask, the result is an all zero IP address which matches the last entry in the forwarding table.
- Basically, the router in the corporate network “punts” and lets the ISP’s routers figure out how to forward the packet.
- Side note: Routers in the real world do not organize forwarding tables the way I described. They use a more efficient search technique called “trie”. You can look up that technique on your own.

Forwarding in the General Internet

- In the previous example, we only had entries in the forwarding table for local networks (or subnets). We used a default route to pass IP packets that had addresses other than local addresses.
- So what does the larger Internet do with those packets? How do they get to their destination?
- We'll discuss a bit here about how forwarding is segmented in the larger Internet.

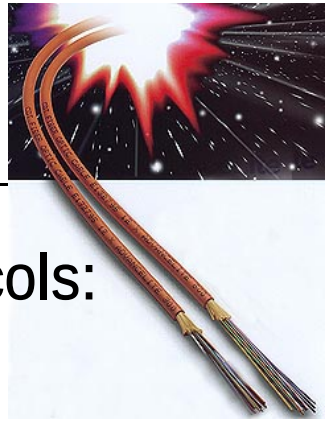
Autonomous Systems (AS)

- An Autonomous System is “a set of routers under a single technical administration, using an interior gateway protocol (IGP) and common metrics to determine how to route packets within the AS, and using an inter-AS routing protocol to determine how to route packets to other ASs” (RFC 4271). For example, a large ISP is an Autonomous System. A large corporate network may, or may not, be an Autonomous System.
- These systems interconnect with each other through peering points, either public or private.

Intradomain and Interdomain

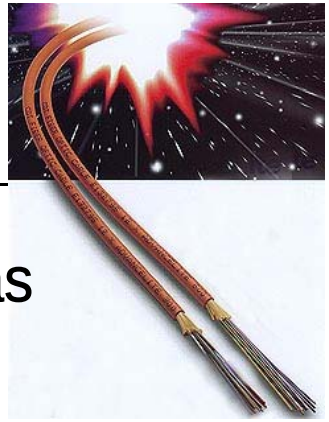
- Routing protocols for intradomain routing are called interior gateway protocols (IGP).
  - The goal, usually, is to find the shortest path.
- Routing protocols for interdomain routing are called exterior gateway protocols (EGP).
  - The goal, usually, is to satisfy the routing policy of the autonomous system.
- We'll look at intradomain routing next, then examine interdomain routing.

Interior Gateway Protocols (IGP)

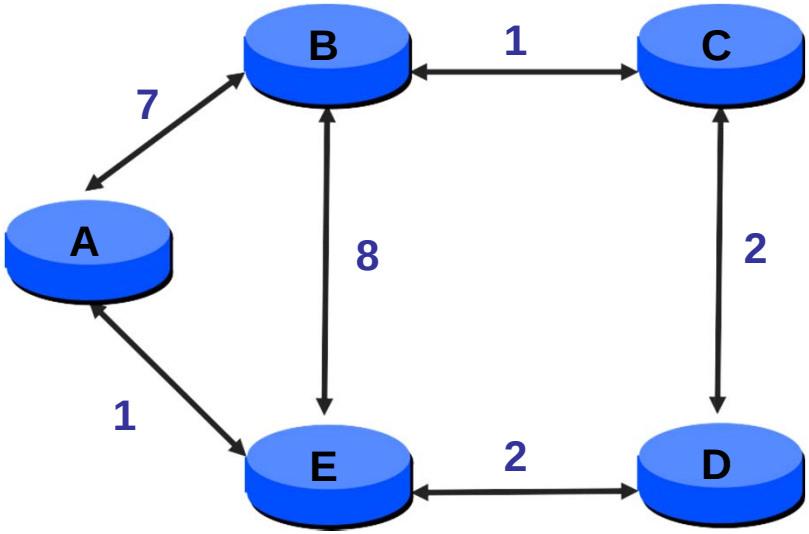


- There are two major classes of interior gateway protocols:
  - Distance-vector protocols.
  - Link-state protocols.
- The term “distance-vector” refers to a group of routing protocols that propagate routing information in the form of a distance measurement (perhaps the number of hops) and the next hop router in that path.
- Each router builds a Forwarding Information Block (FIB) that contains the distance-vector information it knows.
- It then sends this FIB to the routers it’s connected to, and they use that information to update their FIB. And then, they send their FIB to the routers connected to them.

Distance-vector Example



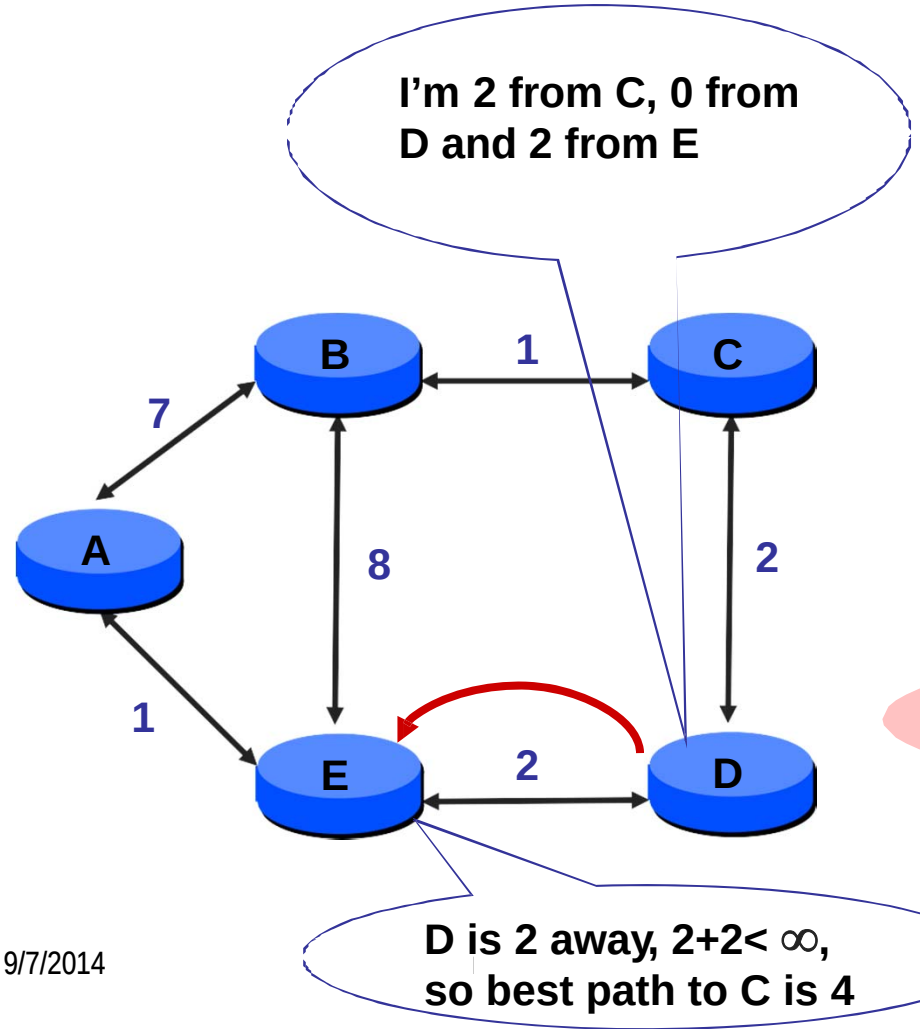
- Before the exchange of information, each node only has information about itself and it's neighbors.
- The table shows the information at each node, so A only knows the distance to itself, B and E.



Info at node	Distance to Node				
	A	**B**	**C**	**D**	**E**
A	0	7	∞	∞	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	∞	2	0

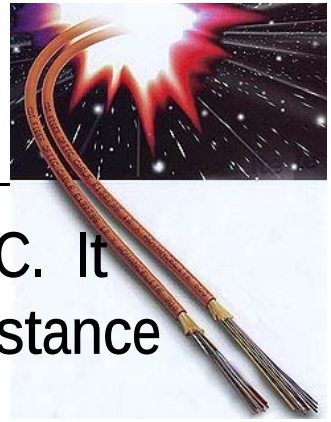
D Sends Vector to E

- When D sends its information to E, E can update its information for the distance to C.

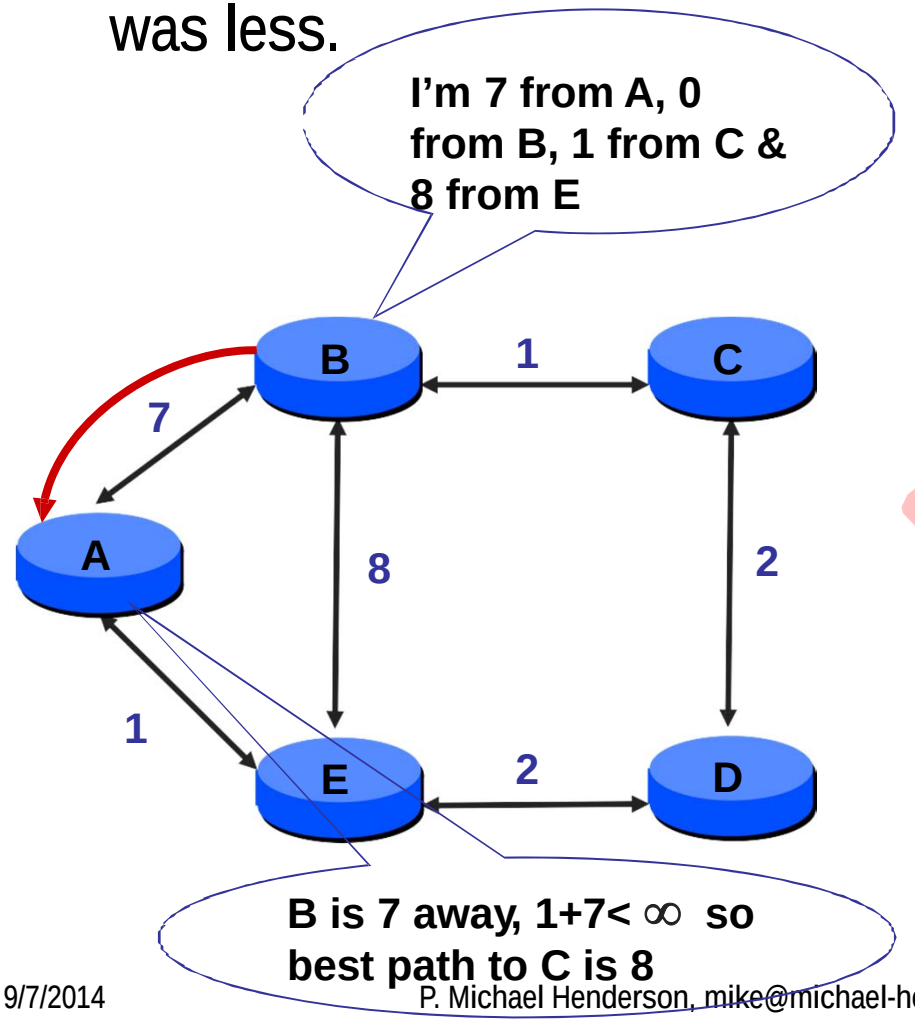


Info at node	Distance to Node A	B	C	D	E
A	0	7	∞	∞	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	**4**	2	0

B Sends Vector to A



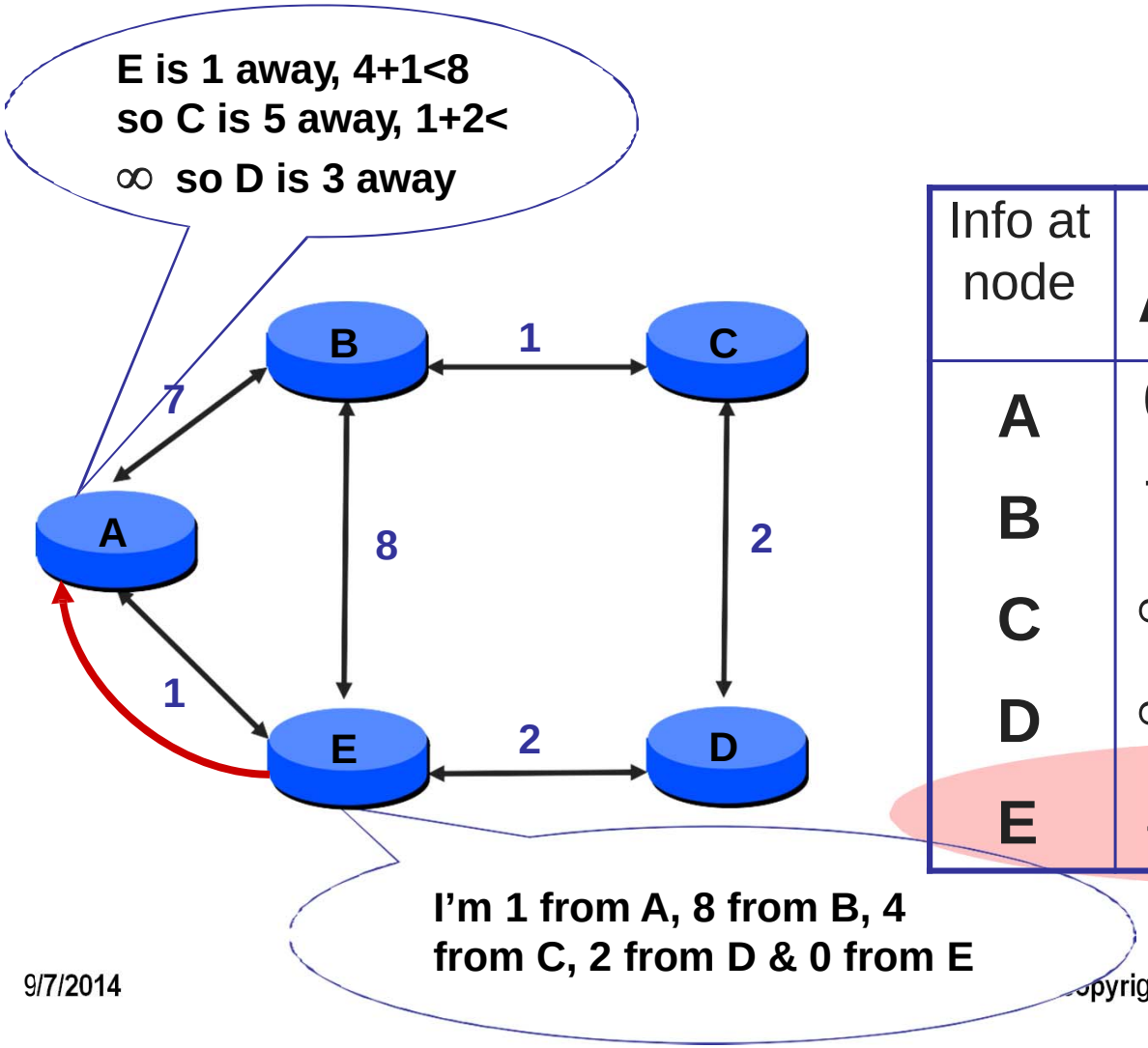
- Since B is 1 away from C, A can update its distance to C. It does not change the distance to E because its initial distance was less.



Info at node	Distance to Node				
	A	B	C	D	E
A	0	7	**8**	∞	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	4	2	0

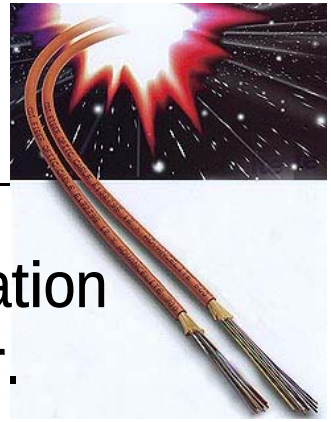
E Sends Vector to A

- A can now update the distance to C and D.

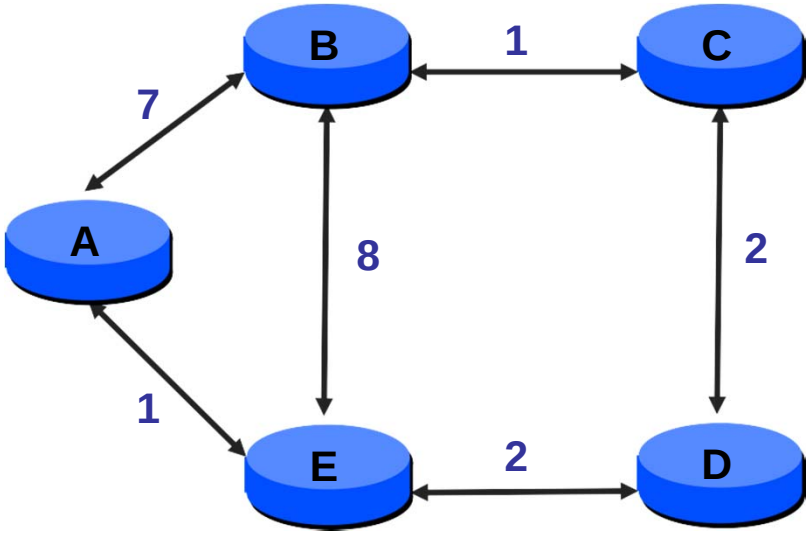


Info at node	Distance to Node				
	A	B	C	D	E
A	0	7	**5**	**3**	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	4	2	0

Continue until Convergence



- After all the routers exchange their Forwarding Information Bases, each will have the distance to each other router.



Info at node	Distance to Node				
	A	**B**	**C**	**D**	**E**
A	0	6	5	3	1
B	6	0	1	3	5
C	5	1	0	2	4
D	3	3	2	0	2
E	1	5	4	2	0

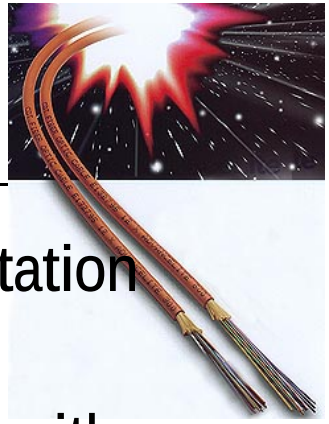
More on Distance-Vector Protocols

- Go to Wikipedia and look up “Distance-Vector Routing Protocol”. They have a good discussion, and examples, of how distance-vector works, and some of the problems of the protocol.

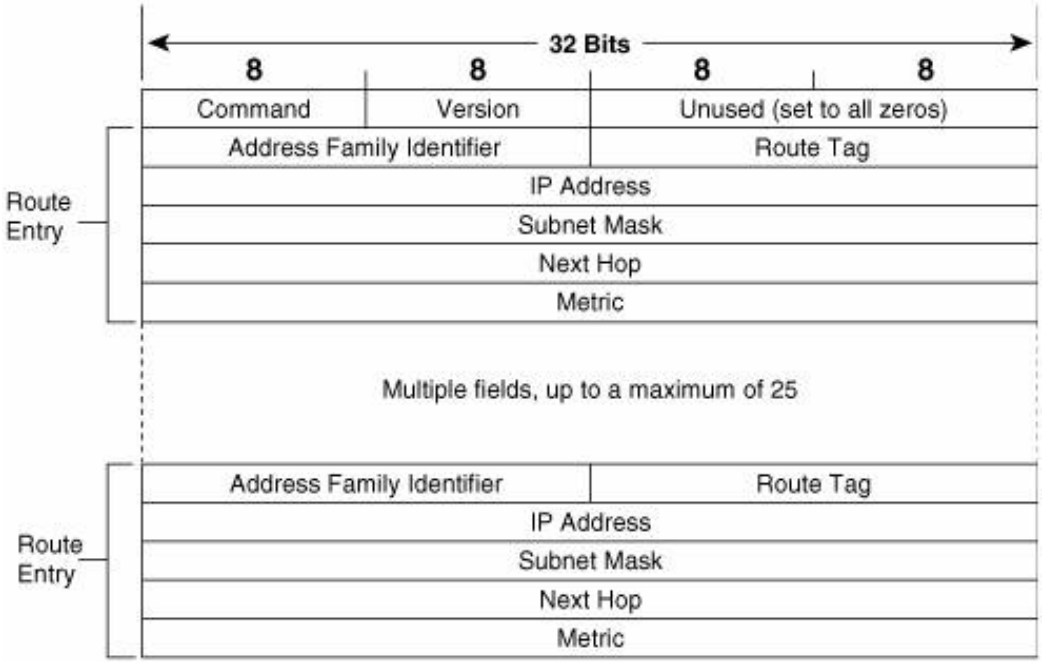
Router Information Protocol (RIP)

- In the examples given, we only saw the paths, but not the destinations (to make the examples simple).
- Let's look at one of the most popular D-V protocols, Routing Information Protocol version 2, and see all of the information sent in an update.
- Note: The other major D-V protocol is Interior Gateway Routing Protocol (IGRP).

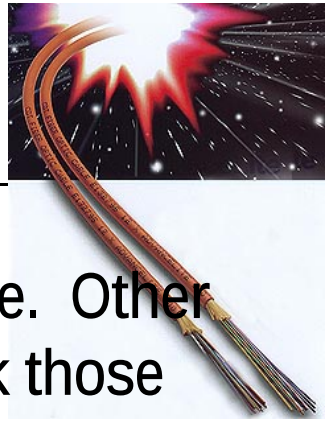
RIP2



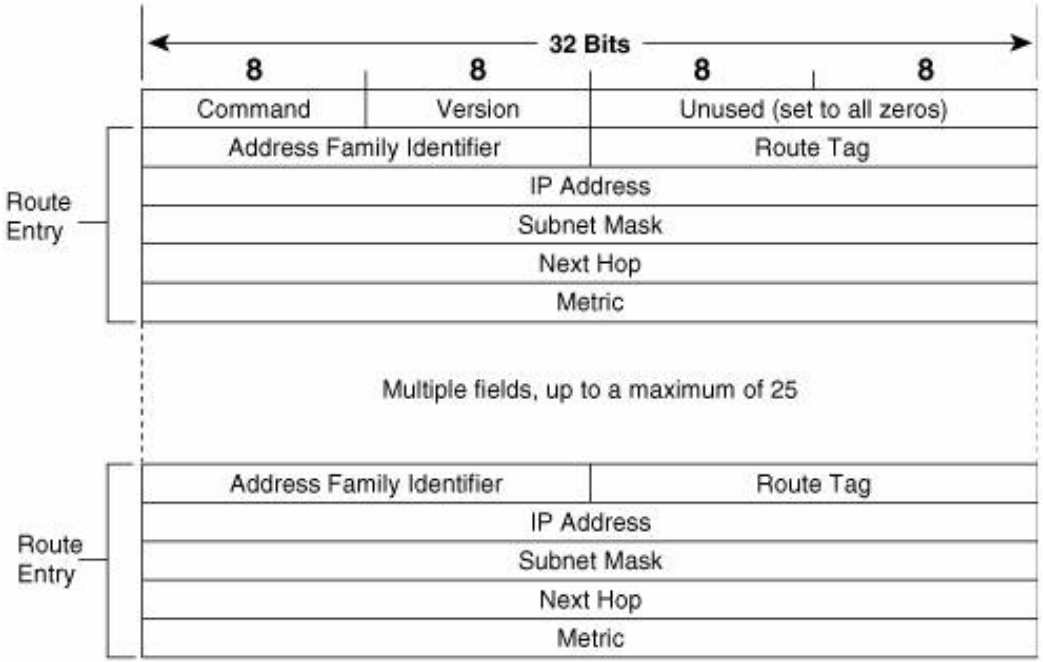
- RIP2 is carried in an UDP packet and uses the destintation port of 520.
- Note that up to 25 networks can be advertised, along with the next router to reach those networks.
- Also note that there is no length indicator for the RIP packet. RIP depends on UDP for the length of the RIP packet.



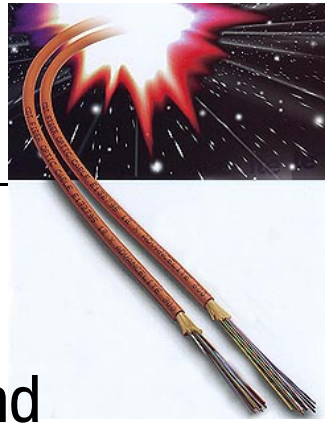
RIP2



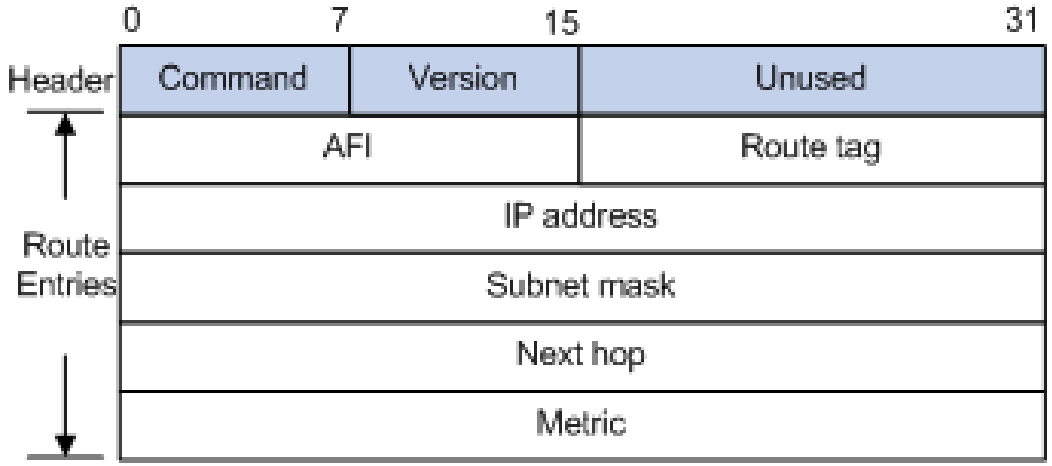
- The most used Command is 2 but others are possible. Other command numbers are 1, 9, 10 and 11. You can look those up.
- Version and Address Family Identifier is 2 for IPV4.
- Route Tag is somewhat complex and not covered here.



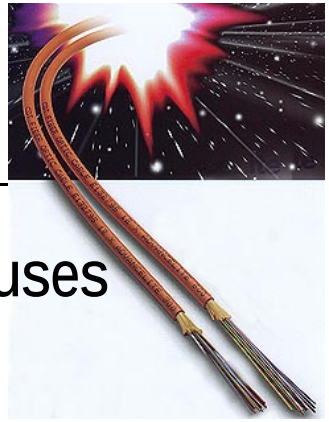
RIPV2



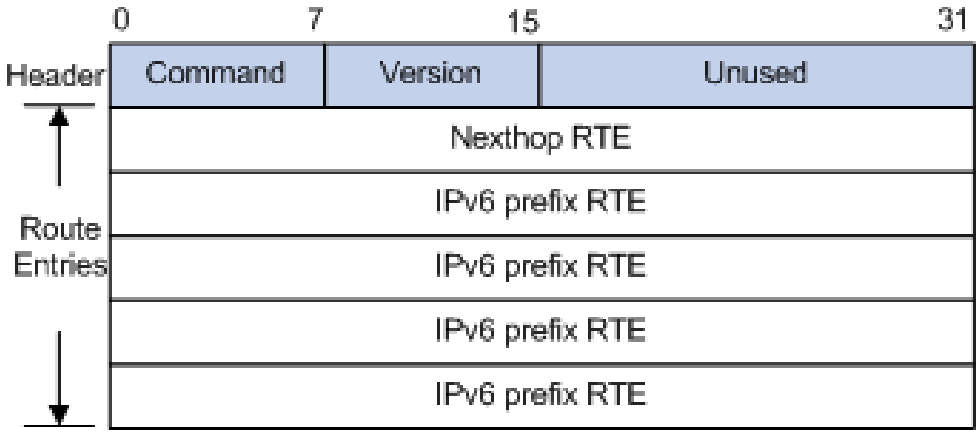
- The IP address is the network address of an attached network.
- The subnet mask is used for classless IP addresses and indicates the part of the IP address that corresponds to the network address.
- Next Hop indicates the address of the “best” next hop to reach the above network.
- The Metric is the hop number, from 1 to 16 (infinity).



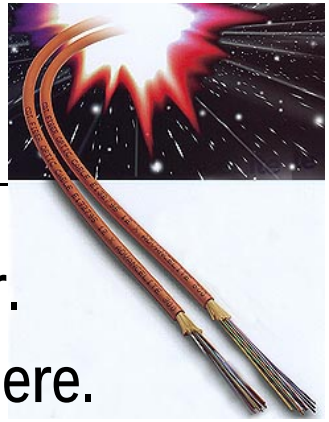
RIPng



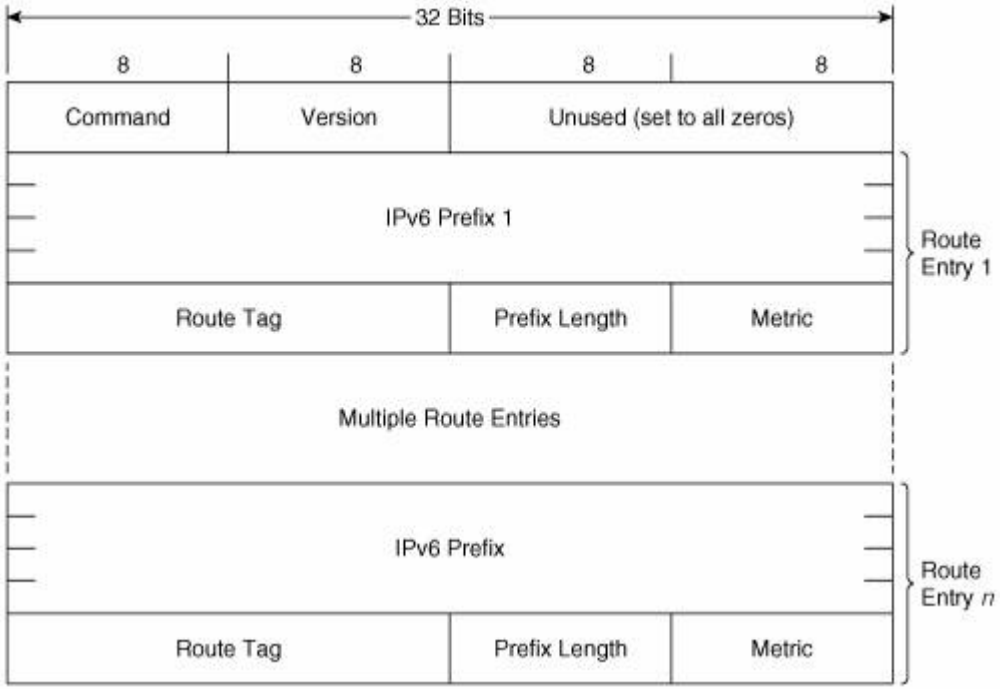
- RIPng is RIP for IPV6. To indicate the new protocol, it uses UDP port 521 (instead of 520).
- Again, no octet count or number of entries. Depends on UDP length indicator.
- Command is 1 (Request) or 2 (Response).
- Version is 1
- RTE is route table entry and is covered next.



RIPng

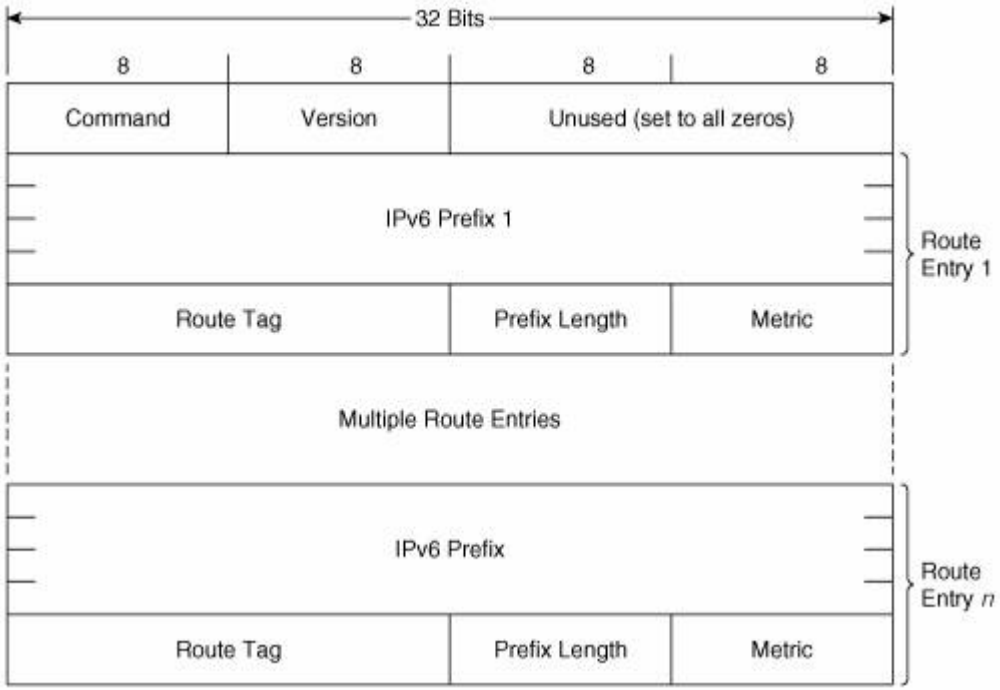


- The first entry is the IP address of the next hop router.
- The Route Tag is the same as IPV4 and not covered here.
- Prefix length is the network part of the IP address, usually 64.
- The Metric is all 1s to indicate next hop router, otherwise the hop count.



RIPng

- The route table entries that follow are the IP addresses of the networks that can be reached through the next hop address.



Distance-vector Routing

- There are a number of issues with distance-vector routing which I'll state without going into detail on each one.
- "Count to infinity" problem. In certain failures, D-V protocols can send information back and forth, increasing the distance each time, until they count to infinity (16 hops).
- Routing loops. A router cannot know if the forwarding path sent to it includes it in the path.
- There are some techniques to mitigate, but not completely solve, these two problems.

Distance-vector Routing

- Distance-vector protocols converge slowly. It can take a long time to get all the forwarding tables up to date after a change to the network (link or node failure – or addition).
- Because a distance of infinity is 16 hops, RIP can only be used on smaller networks.
- RIP generates a lot of supervisory traffic because it periodically propagates it's entire forwarding table to its neighbors.
- Which leads us to Link-state protocols…

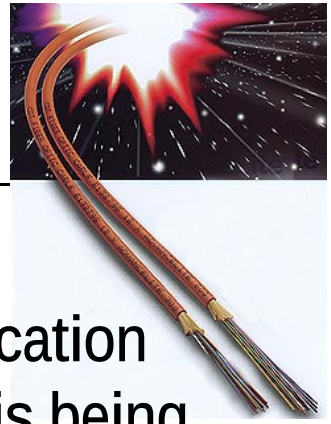
Link-State Protocols

- Link-state protocols provide only:
  - The ID of itself.
  - The ID of its directly connected neighbors with the cost to each neighbor.
  - A sequence number (to make sure the latest packet is used)
  - A time to live for the packet.
- A node learns information about its neighbors by exchanging packets with the neighbors.
- Then, it puts together a Link State Packet (LSP) with the above information and “floods” it on all of it’s links.
- The nodes that receive the LSP forward it on all of their links but keep a record of having receive it. If they receive a duplicate, they discard it.

Link-State Protocols

- Once a node has a number of LSPs, it begins to build a map of the network. Note that it needs an LSP from each node to complete the network map.
  - It looks for LSPs that have it as a neighbor. It can "connect" those nodes to itself.
  - Then it looks for LSPs that have these newly connected nodes as neighbors and "connects" them to those nodes.
  - Eventually, it will build a map of the entire network.
- Once the map of the network is built, it uses Dijkstra's shortest path algorithm to calculate the shortest path to the networks (network IDs) attached to the nodes.
- The two major link-state protocols are Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS). We'll look at OSPF.

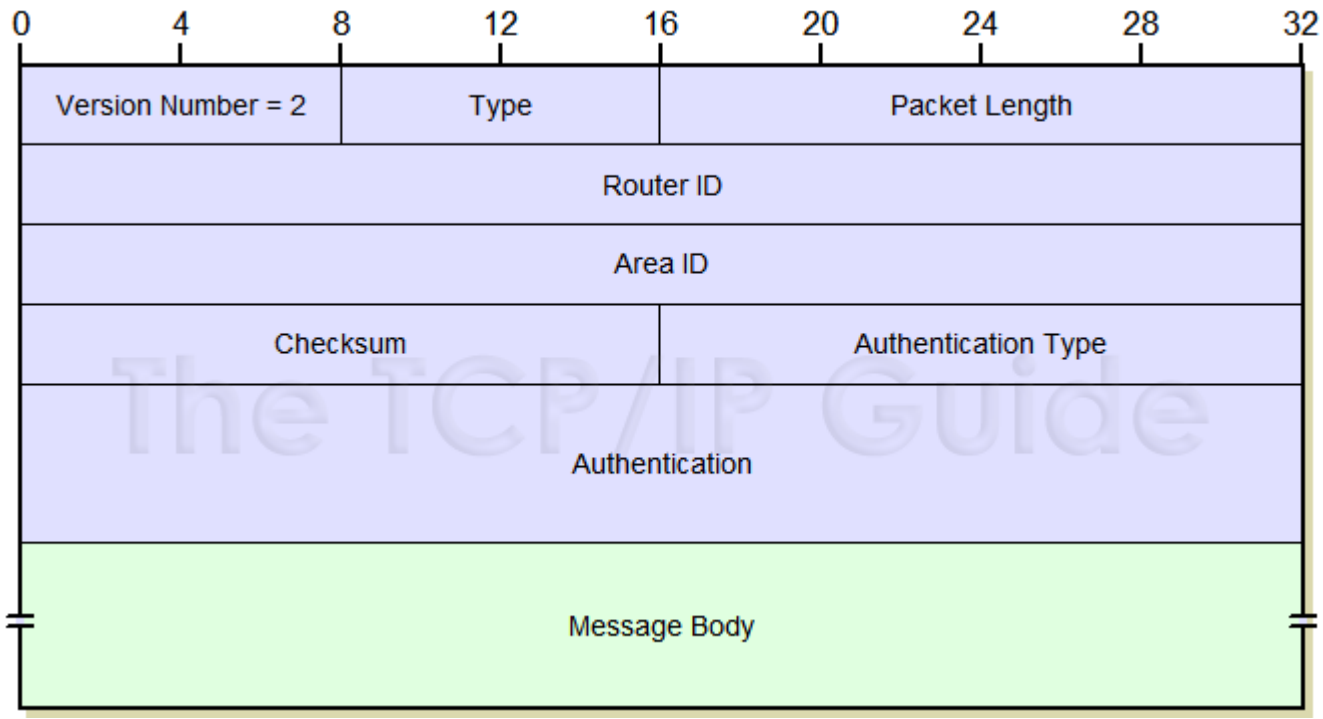
Open Shortest Path First (OSPF)



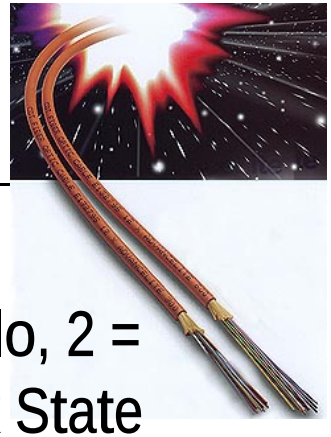
- The “Open” in OSPF refers to the fact that the specification is an open, non-proprietary standard, not that a path is being “opened”.
- The “Shortest Path First” part of the name refers to the alternate name for link-state routing.
- OSPF is carried within an IP packet – it is not carried by UDP or TCP – with protocol number 89. OSPF implements its own error detection and correction techniques. It is a layer 3 protocol in its own right.

Open Shortest Path First (OSPF)

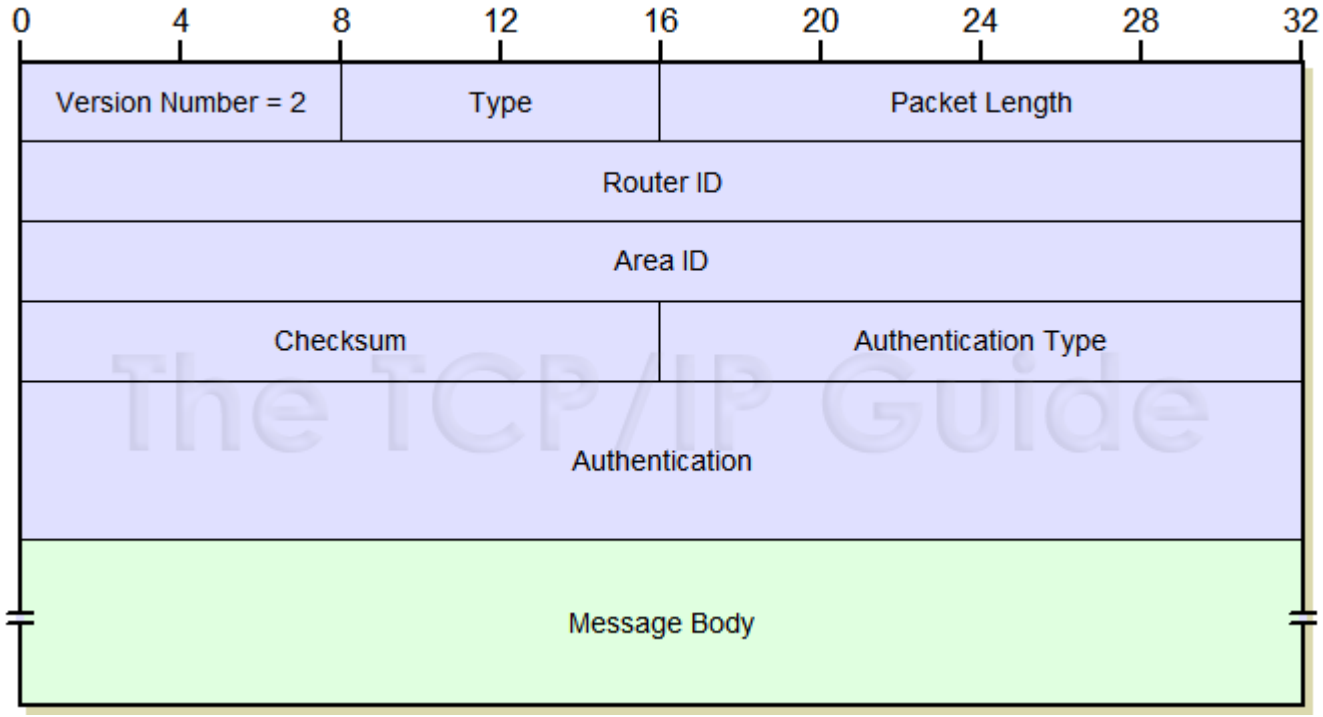
- We'll look at the OSPF protocol next.
- OSPF has a common header and different data components. We are only going to look at the basics of OSPF here.
- This is the LSP header. I'll explain the body later.



Open Shortest Path First (OSPF)

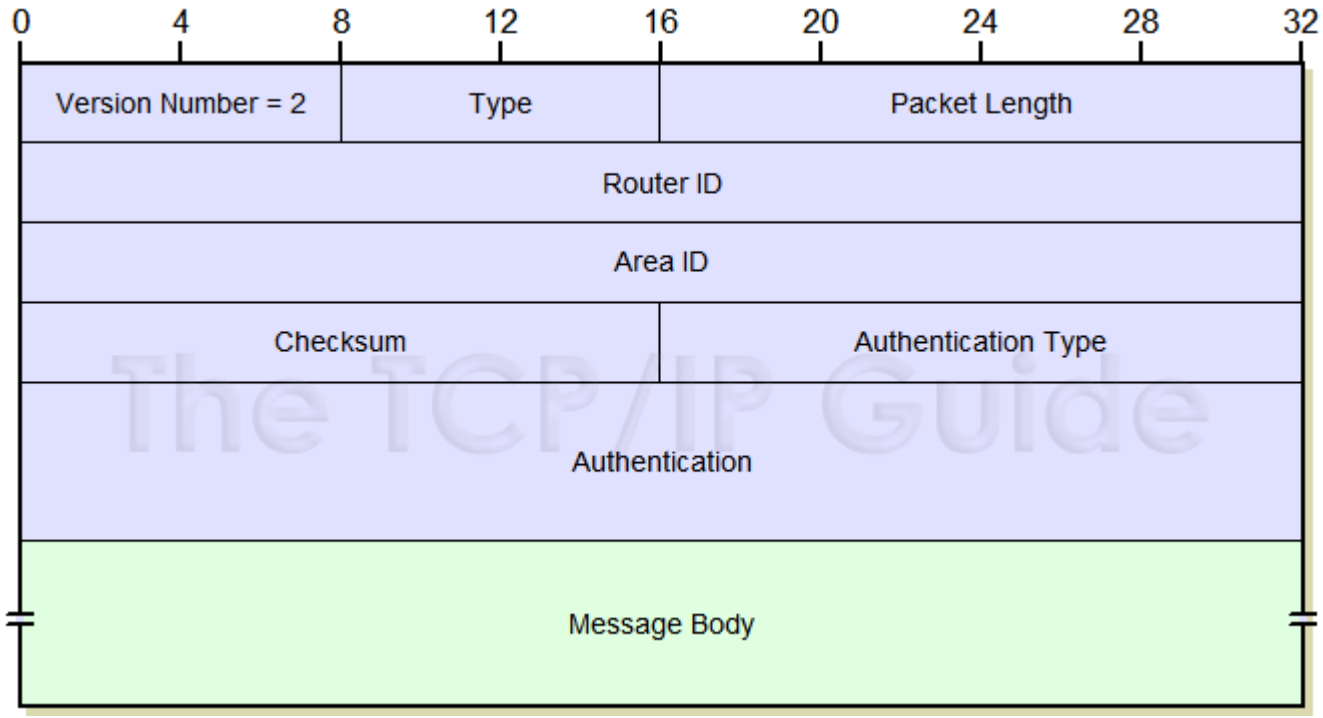


- The Version number is 2 to indicate OSPFv2.
- The type indicates the type of OSPF message: 1 = Hello, 2 = Database description, 3 = Link State Request, 4 = Link State Update, 5 = Link State Acknowledgement.
- Packet length is the total length of the packet, including this header.



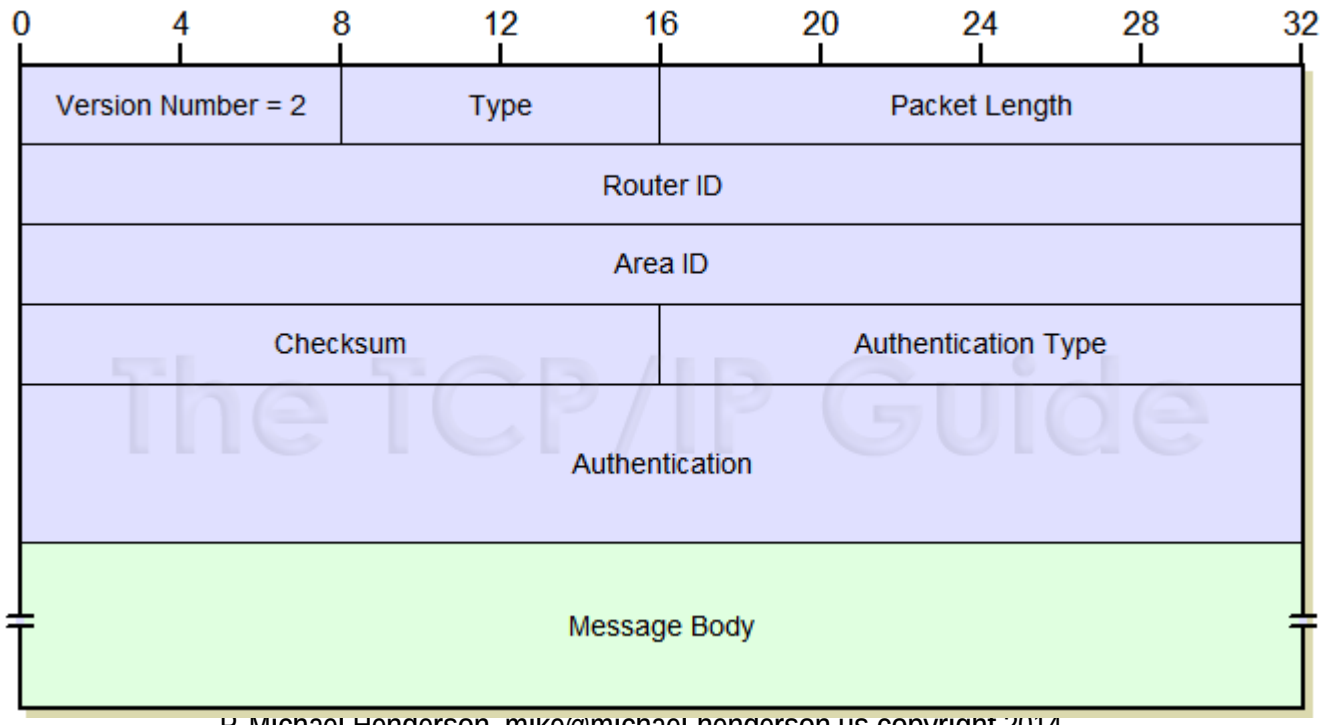
Open Shortest Path First (OSPF)

- Router ID is the IP address of the router sending the LSP.
- I won't go into the Area ID.
- Checksum is calculated the same as the IP checksum.
- Authentication type is: 0 = no authentication, 1 = password, 2 = cryptographic authentication.



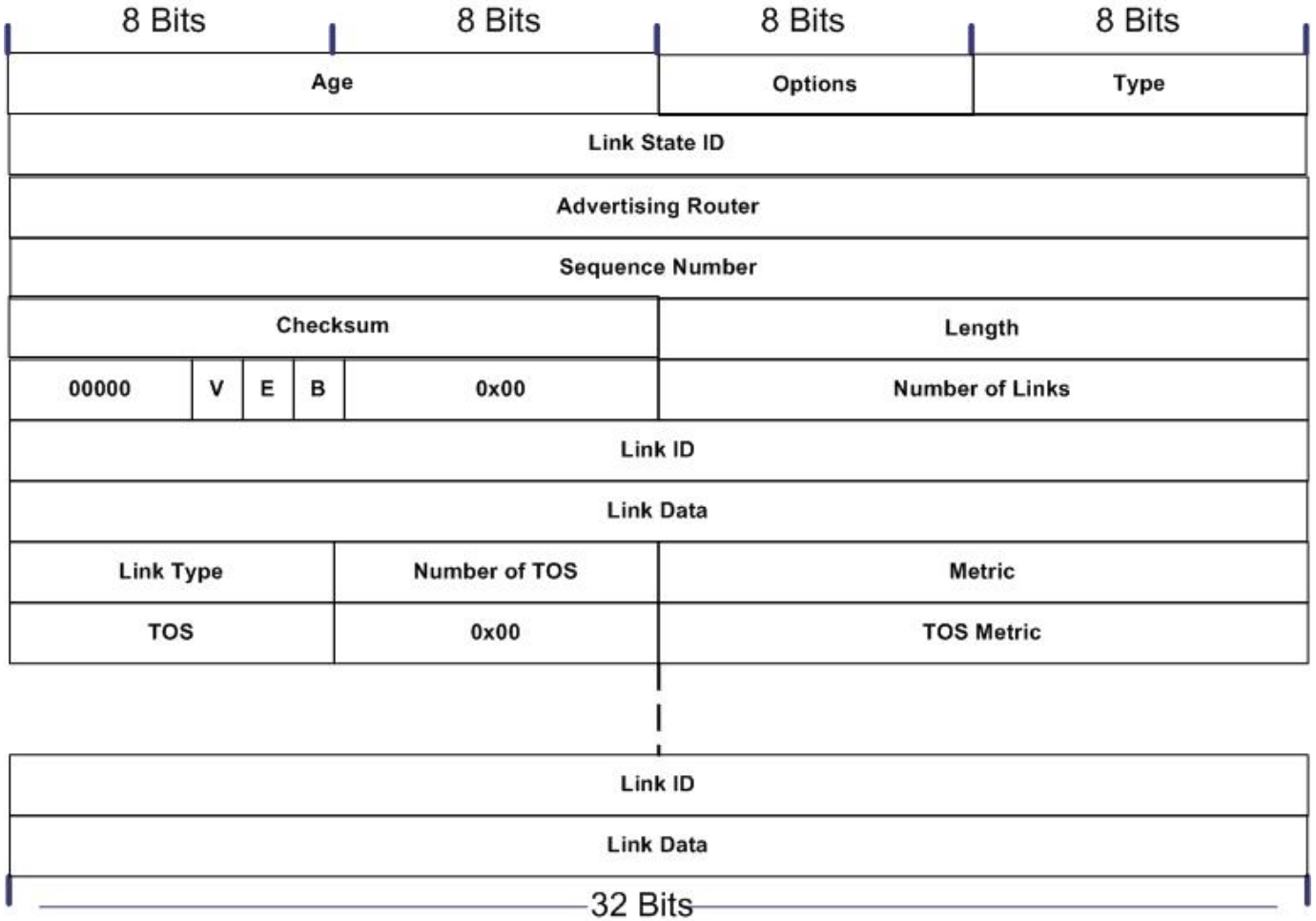
Open Shortest Path First (OSPF)

- Authentication is a 64 bit field to be used as required by the authentication.

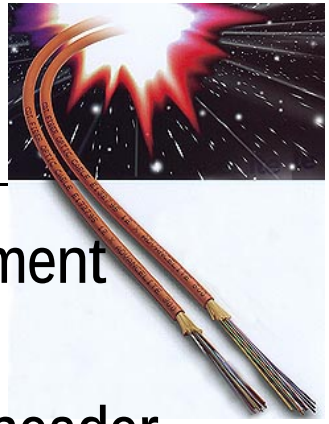


OSPF Link-State Advertisement

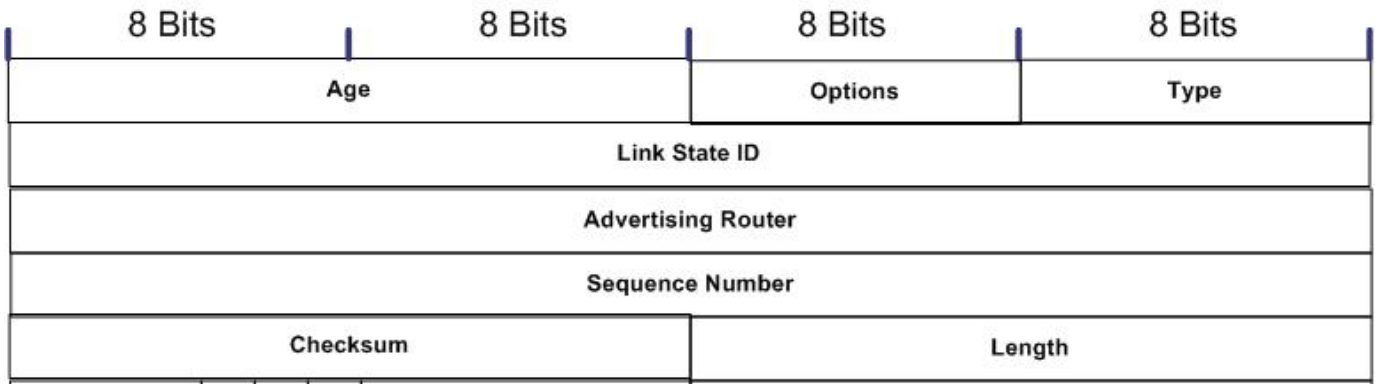
- Certain of the Link-state message types will have a Link-state advertisement after the header.
- The first 20 octets of this message is the advertisement hdr.



OSPF Link-State Advertisement

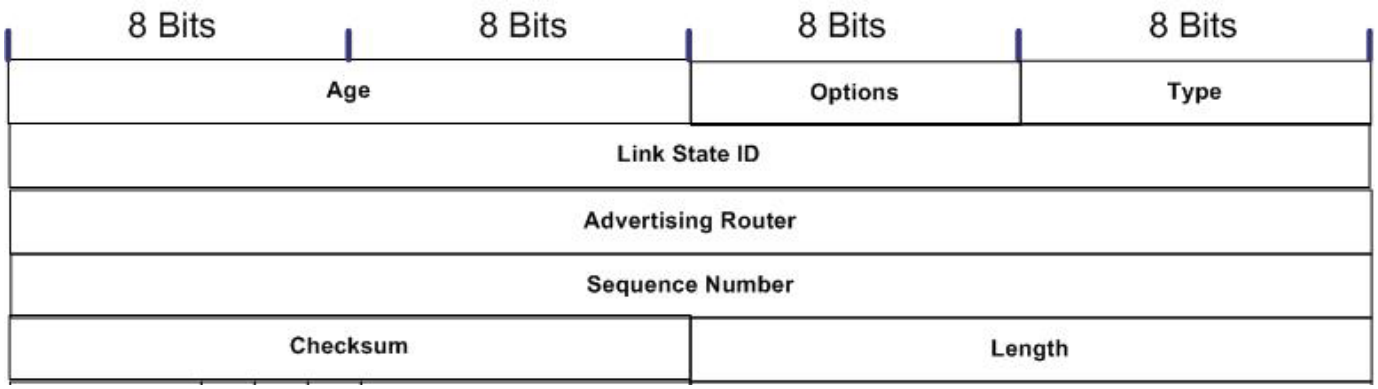


- The Age is the time in seconds since the LS advertisement was generated.
- Options refers to options in the service following this header.
- Type is type of advertisement: 1 = router links, 2 = network links, 3 = summary link (IP), 4 = summary link (autonomous system boundary routers), 5 = Autonomous system external link.
- Link State ID is the portion of the network described by the LS advertisement.



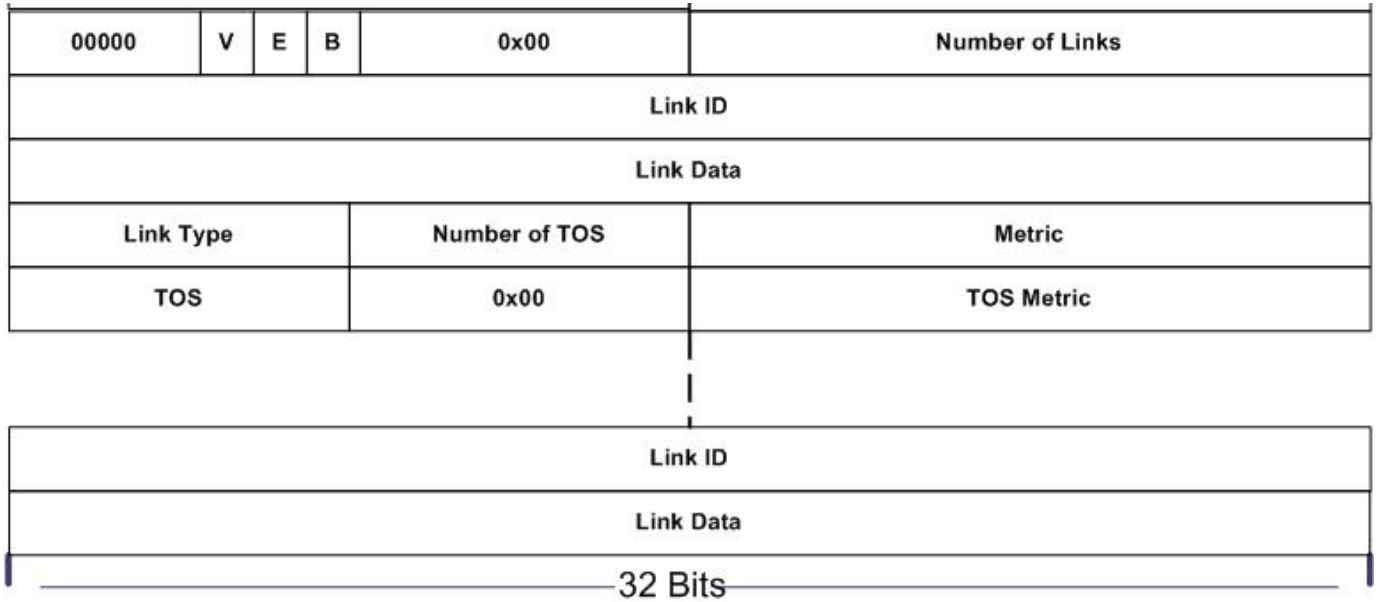
OSPF Link-State Advertisement

- The Advertising Router is the IP address of the router generating this LS advertisement.
- Sequence Number is allow the receiver to know which advertisement is the latest.
- Checksum is the standard Internet checksum.
- Length is the total length in octets of the LS advertisement.
- Now, let's look at the actual advertisement.



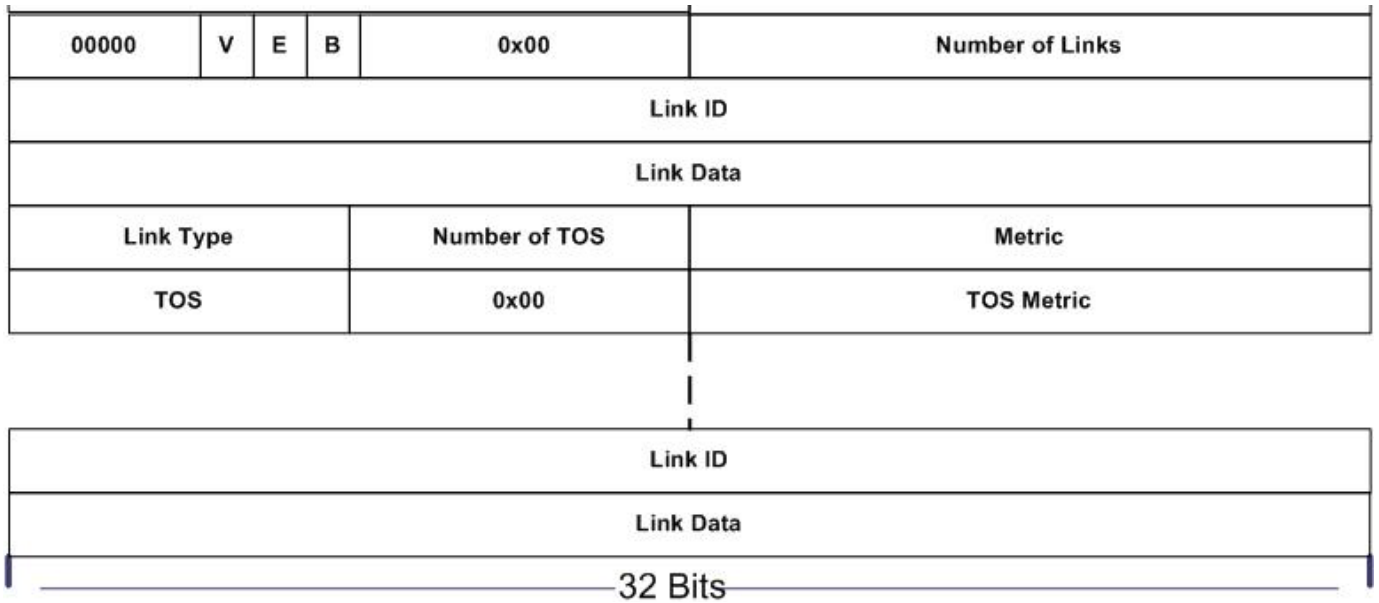
OSPF Link-State Advertisement

- The V bit is the Virtual Link Endpoint bit and will not be further described here.
- The E bit is set to 1 when the router is an autonomous system boundary router.
- The B bit is set to 1 when the router is an area border router.
- #Links is the number of entries that follow.

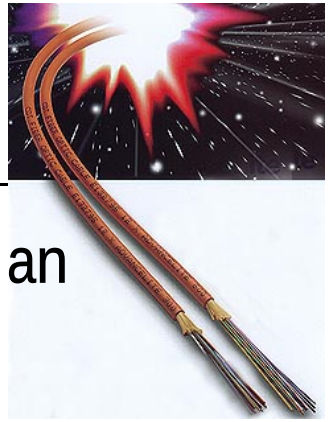


OSPF Link-State Advertisement

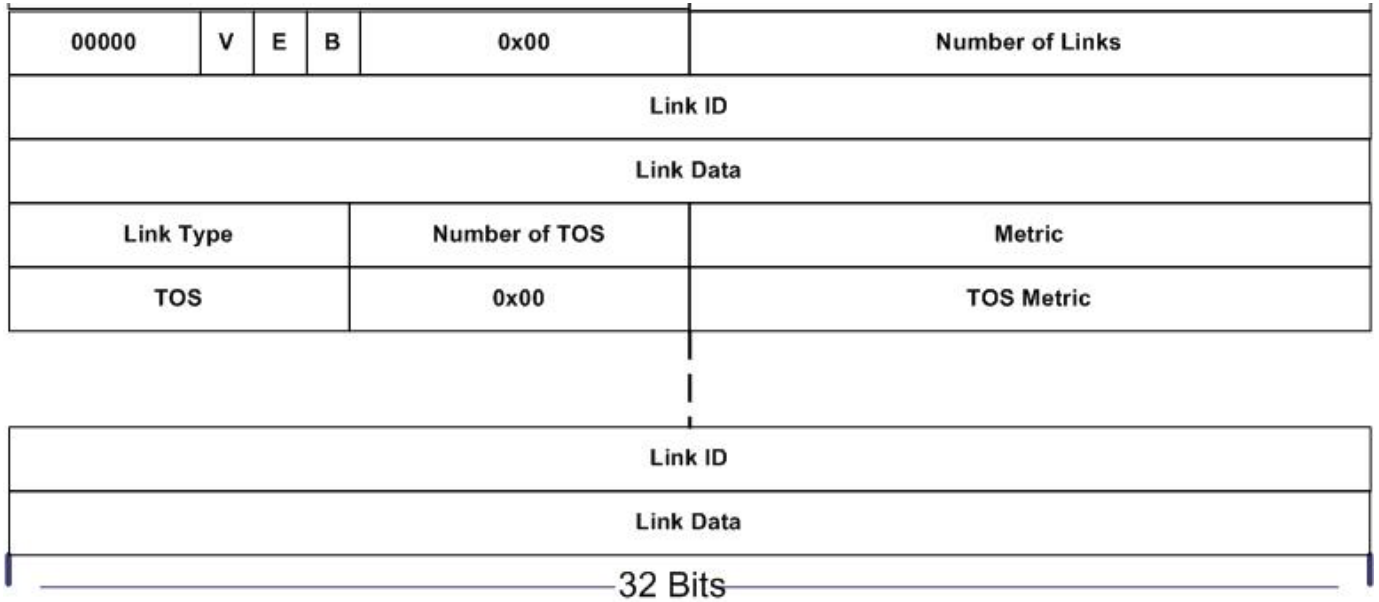
- Jumping down to the Link Type, it can have values as follows: 1 = point-to-point connection to another router, 2 = connection to a transit network, 3 = connection to a stub network, and 4 = virtual link.
- The Link ID and Link Data fields depend on the Link Type and I won't detail them here, except to say that for a Link Type=3, the Link ID is the IP Network address of the attached network and the Link Data is the network mask.



OSPF Link-State Advertisement



- The Number of TOS is for backward compatibility with an earlier version of ODPF.
- Metric is a cost of the link. Lower is better.
- TOS indicates what criteria to use for computing the link: 0 = normal service, 2 = minimize monetary cost, 4 = maximize reliability, 8 = maximize throughput, 16 = minimize delay.
- TOS Metric is the metric associated with the TOS.



Distance-vector (DV) vs Link-State (LS)

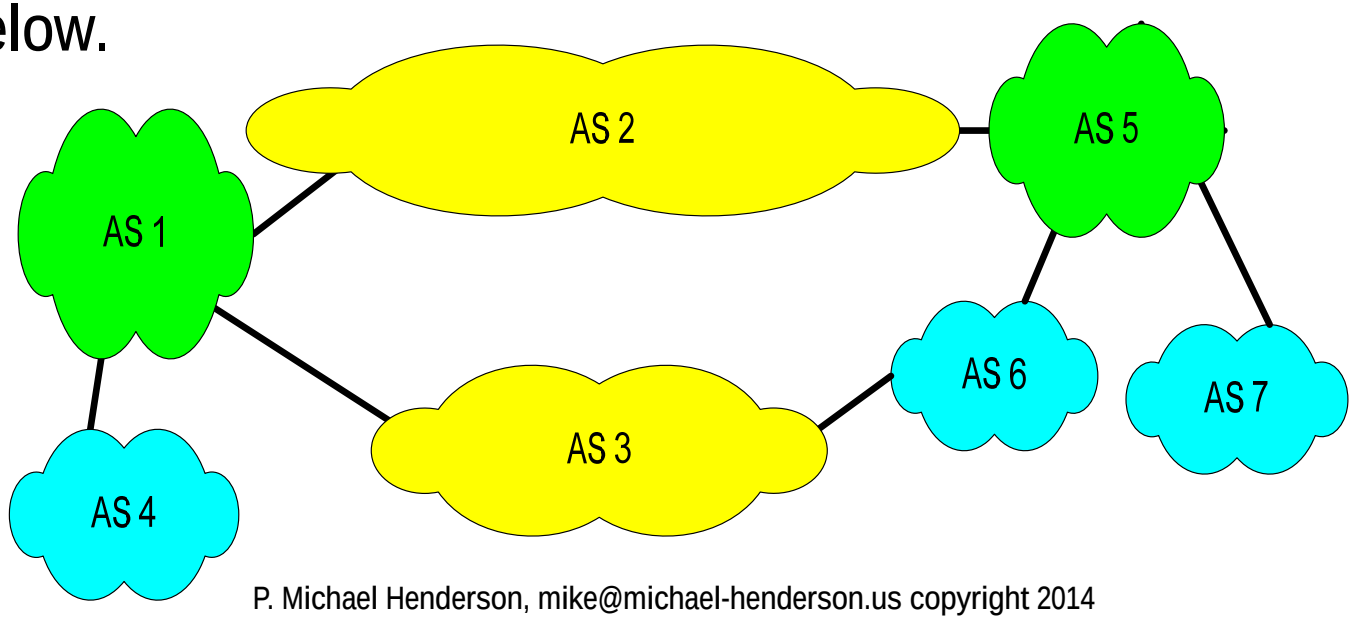
- Since LS floods its messages, more messages are sent than with DV.
- LS converges much faster than DV. Additionally, DV can have routing loops, and has the count to infinity problem.
- If a router malfunctions, LS can send an incorrect link cost. But each router computes its own forwarding table.
- If a router malfunctions, DV can advertise an incorrect path cost. Each router's table is used by all the other routers, which means the error propagates.
- RIP (DV) has an infinity count of 16, limiting network size.
- LS protocol is more complex to administer than DV.
- LS winds up being used in large autonomous networks, DV in smaller networks.

Problems with Interior Gateway Protocols

- No routing update protocol can scale to the whole Internet.
  - Every router in the Internet would have to exchange routing information.
  - Routing update traffic would swamp any other traffic.
- There's a practical limit to how many routers can participate in any IGP.
  - Beyond a certain limit, the convergence time – the time it would take for all the routers to be informed of a change – would be too great to be practical.
  - During that update time, data packets are likely to be sent to the wrong destinations, or dropped.
- To solve this problem, we introduce the concept of autonomous systems, and a two level abstraction.

Interdomain Routing

- The Internet is a two level network. At the lowest level we have networks which use intradomain routing protocols, as we just looked at.
- The next level up is autonomous systems – networks that use intradomain routing protocols.
- When we look at this second level, we see groups of networks that are connected by routers, like the figure below.

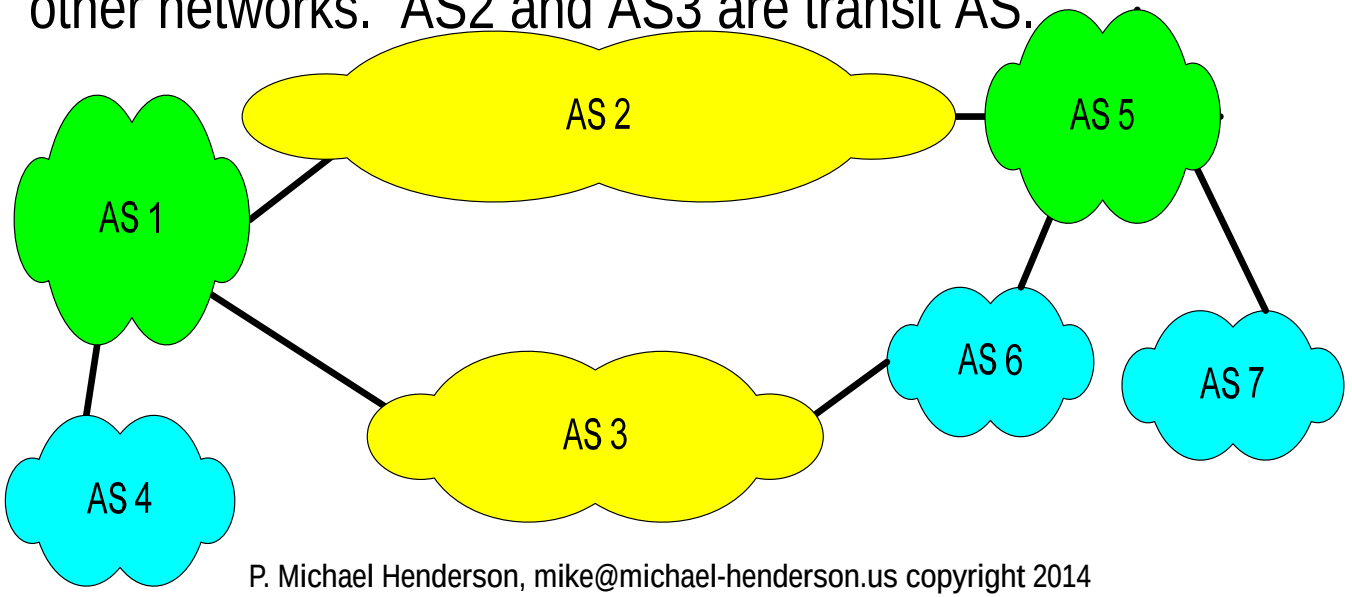


Autonomous Systems (AS)

- Each Autonomous System (AS) has an Autonomous System Number (ASN) issued by the Internet Assigned Numbers Authority (IANA).
- At first, these were 16 bit numbers but with the growth of the Internet, they are now 32 bit numbers. The 32 bit numbers are usually expressed as x.y, where x and y are the decimal value of each 16 bits.

Autonomous Systems (AS)

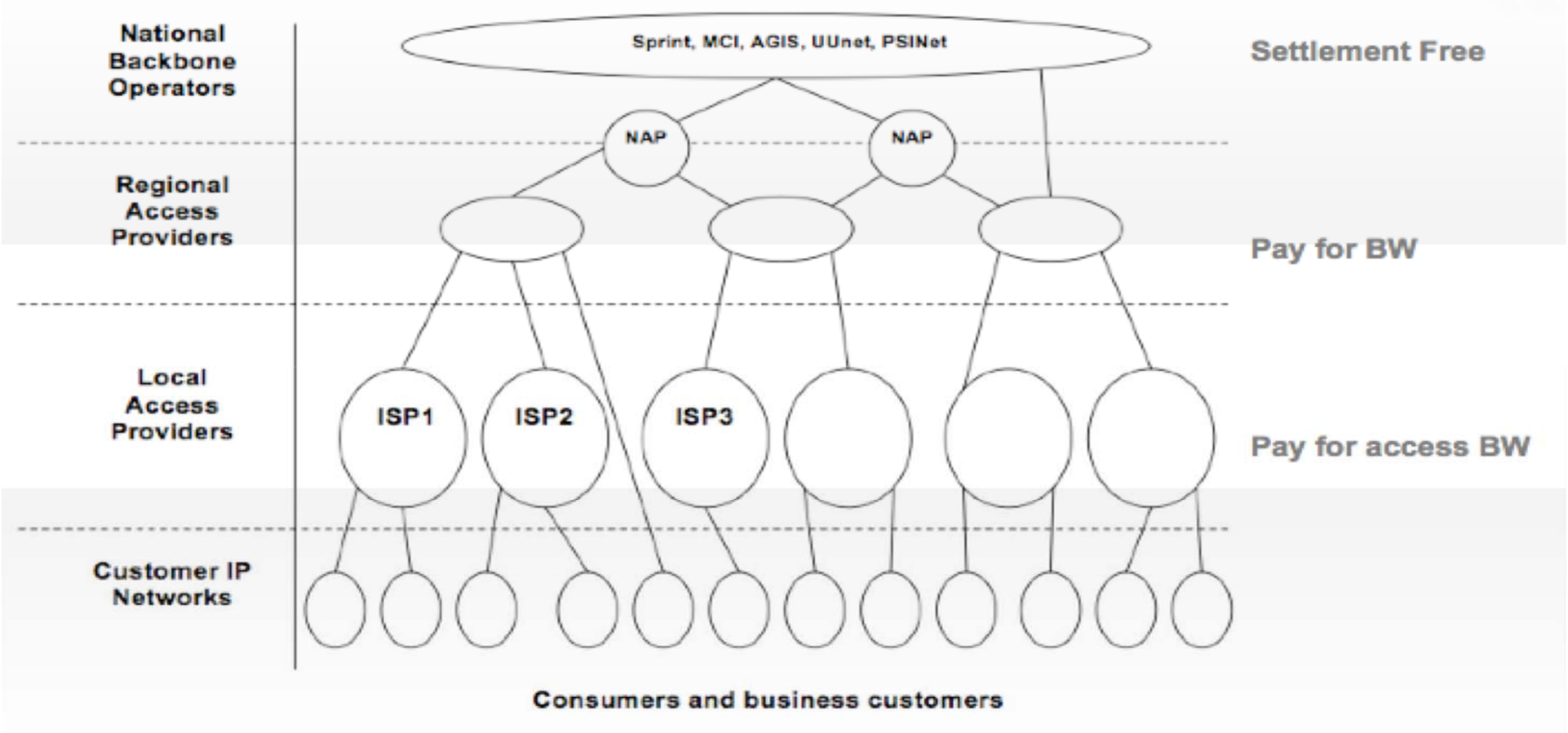
- There are three types of autonomous systems.
  - A mulithomed AS is one that has connections to multiple other AS, but does not provide transit. AS1, AS5 and AS6 are multihomed AS.
  - A stub AS is one that is only connected to one other AS. AS4 and AS7 are stub AS.
  - A transit AS is one that provides connections through itself to other networks. AS2 and AS3 are transit AS.



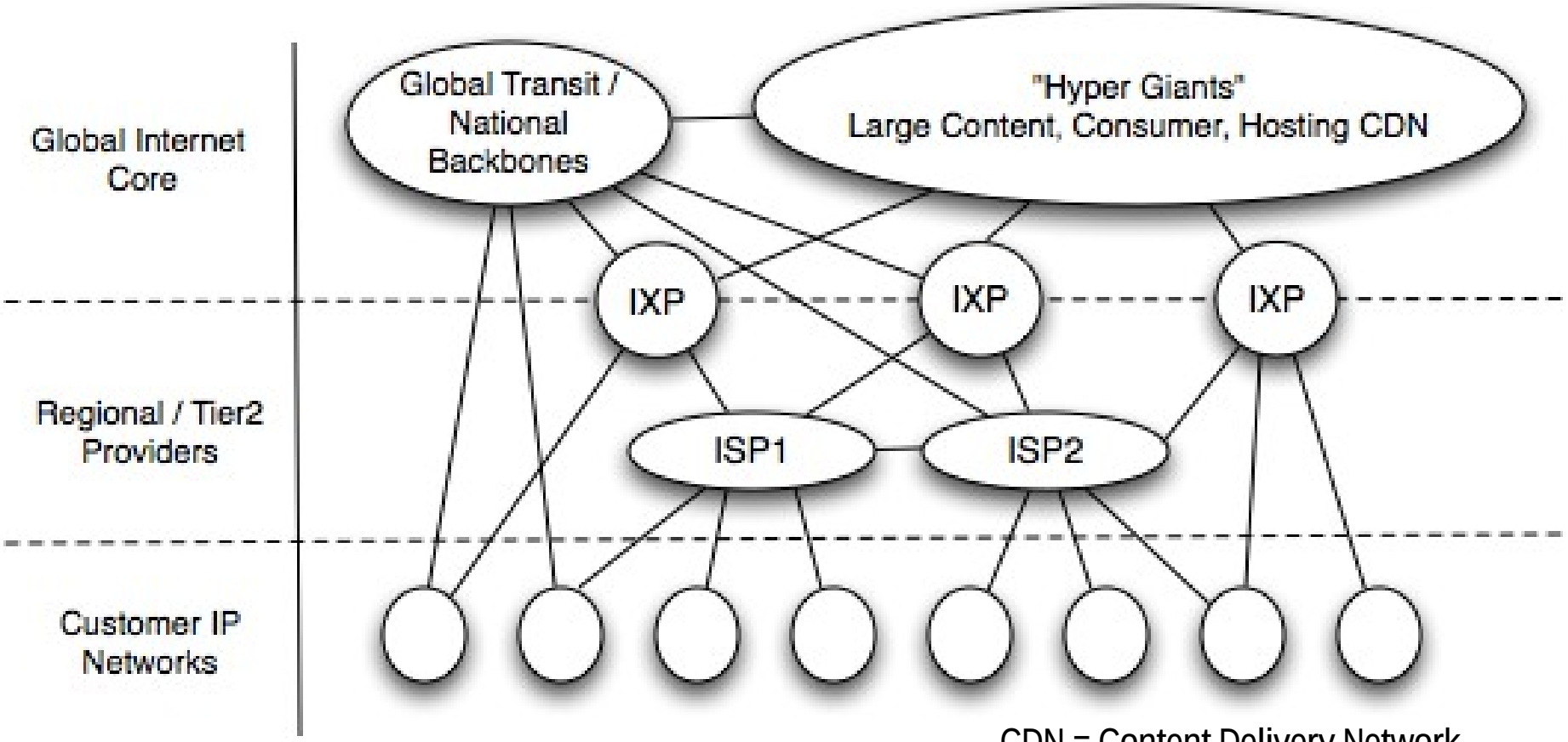
Autonomous Systems

- Autonomous systems may be of different technologies, but most important , they are under different administrative authority.
  - Most autonomous systems are Internet Service Providers (ISPs).
  - They usually want to control what traffic comes through their network.
  - And what traffic is sent through other networks, and where that traffic is handed off to another provider.

Traditional View of the Internet



The Internet Today



CDN = Content Delivery Network
IXP = Internet Exchange Point

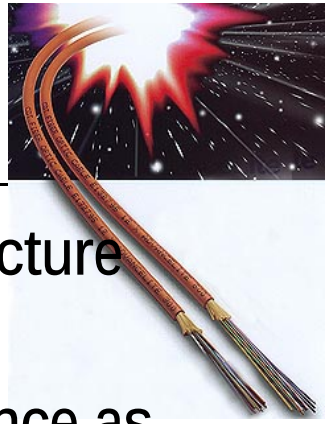
Autonomous Systems

- There is a hierarchy of AS.
  - Large, tier-1 provider with a nationwide backbone.
  - Medium-sized regional provider with smaller backbone.
  - Small network run by a single company or university.
- The major tier 1 providers are: AOL, AT&T, Global Crossing, Level3, UUNET, NTT, Qwest, SAVVIS (formerly Cable & Wireless), and Sprint.
  - These have full peer-to-peer connections between them.
  - Have national or even international backbone.

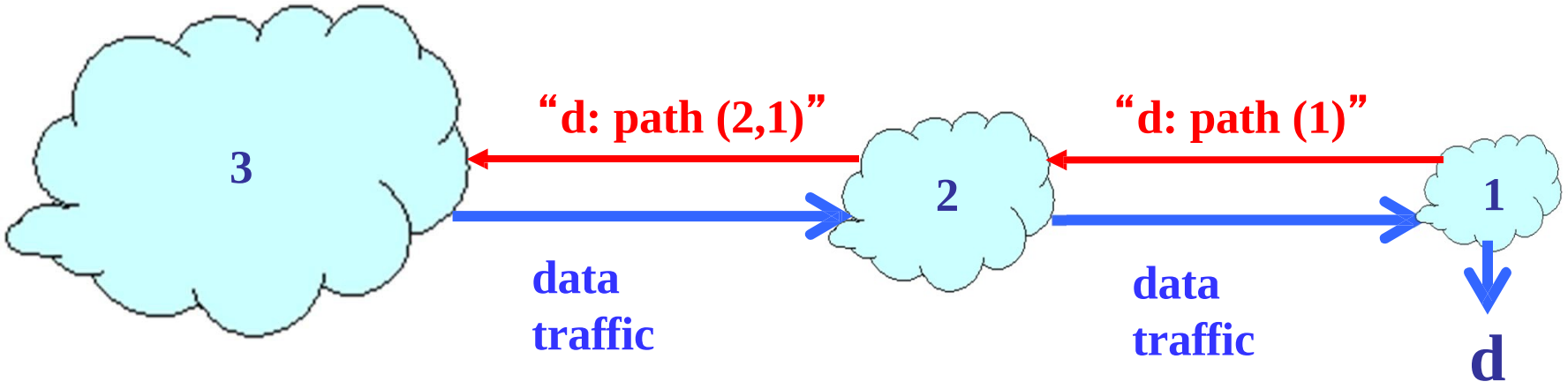
Exterior Gateway Protocol

- An Exterior Gateway Protocol is used to pass “network reachability” information between autonomous systems.
- The protocol used today is Border Gateway Protocol, version 4, commonly referred to as BGP.
- Two autonomous systems connect by providing a communications link between a router in each system, each router running BGP.
- BGP advertises “reachability” instead of routing and uses a “path-vector” protocol.
- BGP is carried by TCP so that communication is reliable.

Path-Vector

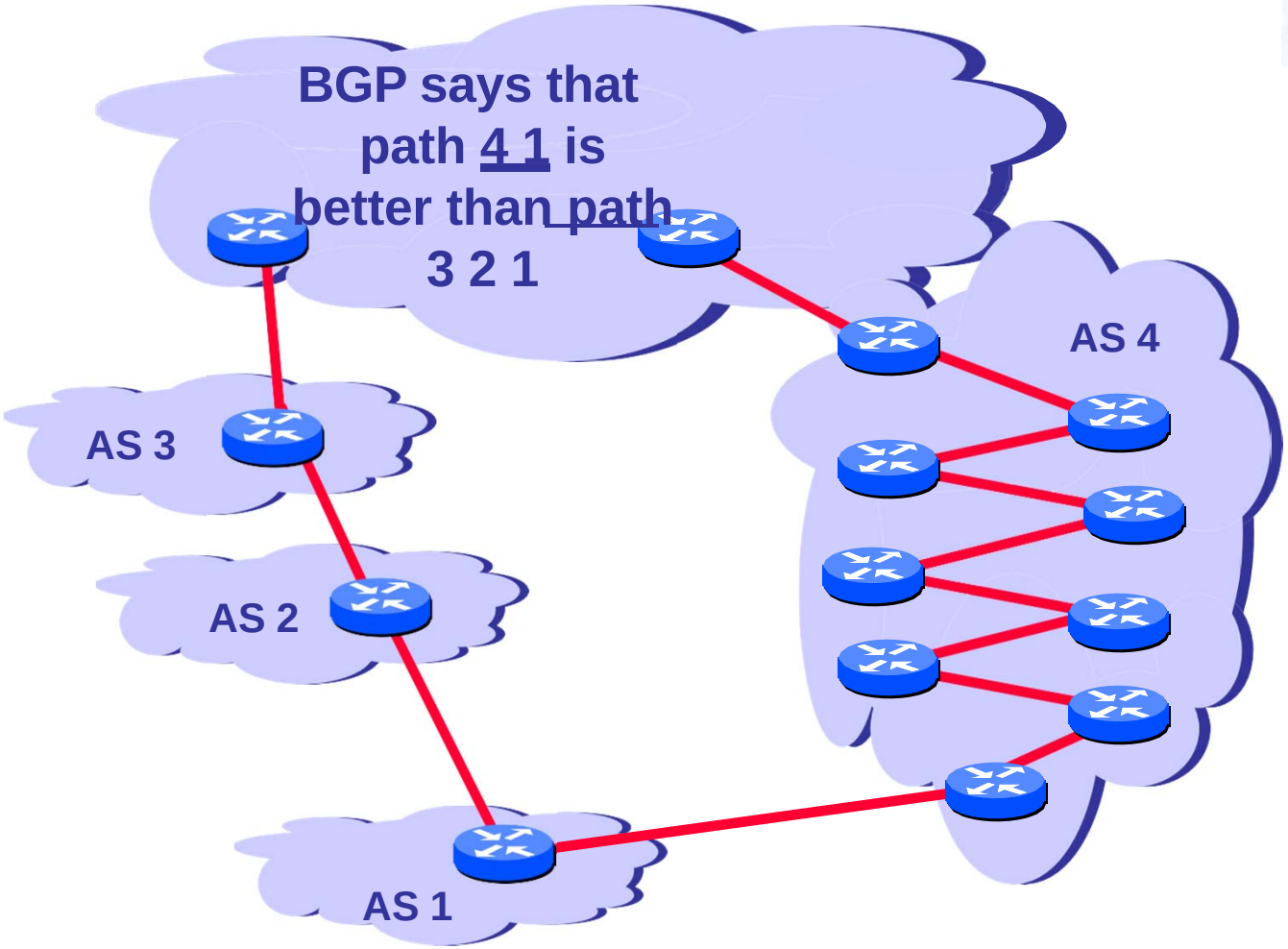


- Path is from AS to AS, with no view of the internal structure of the various AS.
- Sends the actual path from AS to AS, rather than distance as in DV protocols.
- Avoids routing loops and the “count to infinity” problems of DV.
- Supports “policies” of the AS. For example, can force certain traffic along paths chosen by the administration.



Path-Vector

- May not wind up with the “shortest” path.



Problems with BGP

- Routing can be unstable during the time it takes to update routes (converge) when a link goes down. Packets can be sent to wrong destinations or dropped.
- Even with route aggregation, a backbone router can have 400,000 forwarding entries.
- BGP does nothing to balance the traffic load between autonomous systems.

Name Server

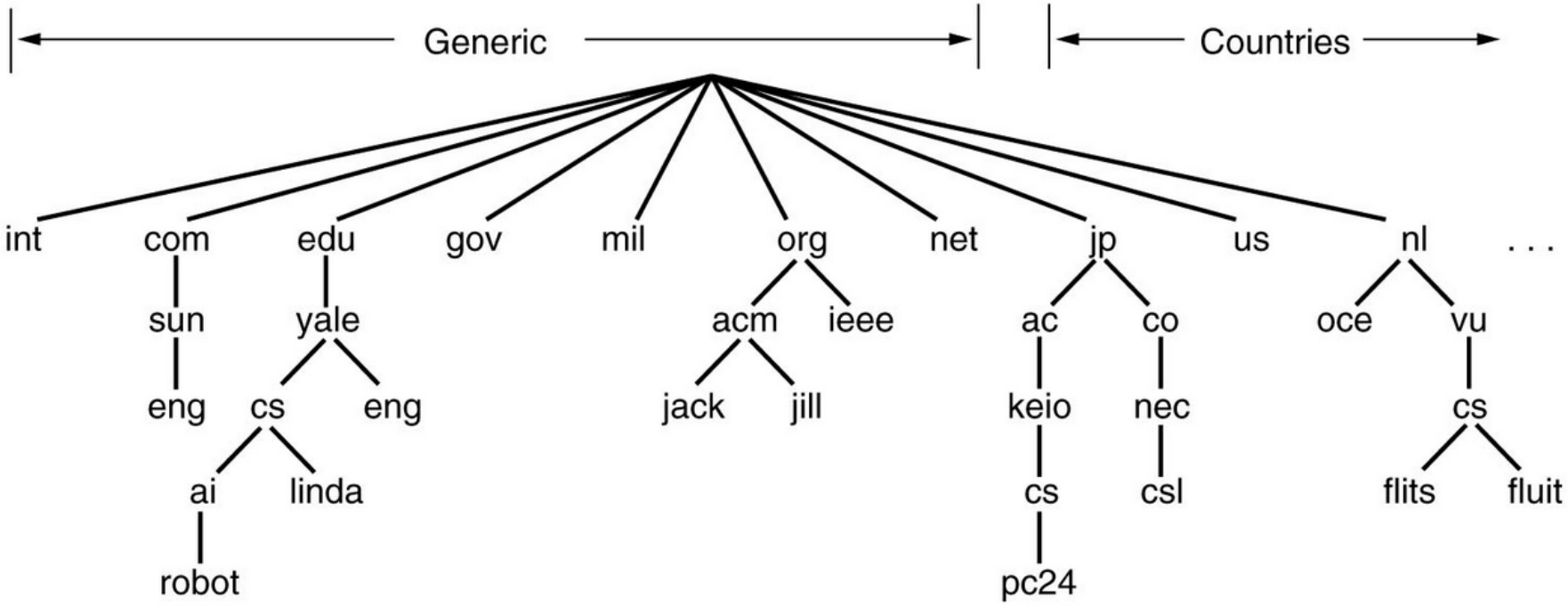
- Up to this point, all we've talked about is IP addresses. But you know that when you access a web site, you do so by name, not by IP address. How do those names get converted to IP addresses?
- The answer is through a "name server".
- But first, let's talk about what an Internet name is.

Names in the Internet

- An Internet name, called a “Domain Name”, can be made up the 26 letters of the alphabet, plus the numbers 0 to 9, plus the hyphen (although it may not start or end with a hyphen).
- A domain name is a hierarchy of names, separated by the period character (called a “dot”), such as example.com
- Domain names are case insensitive. So TheExample.com is the same as theexample.com
- The levels of the hierarchy extend from right to left. Thus, for example.com, com is the highest level and example is a subordinate level.

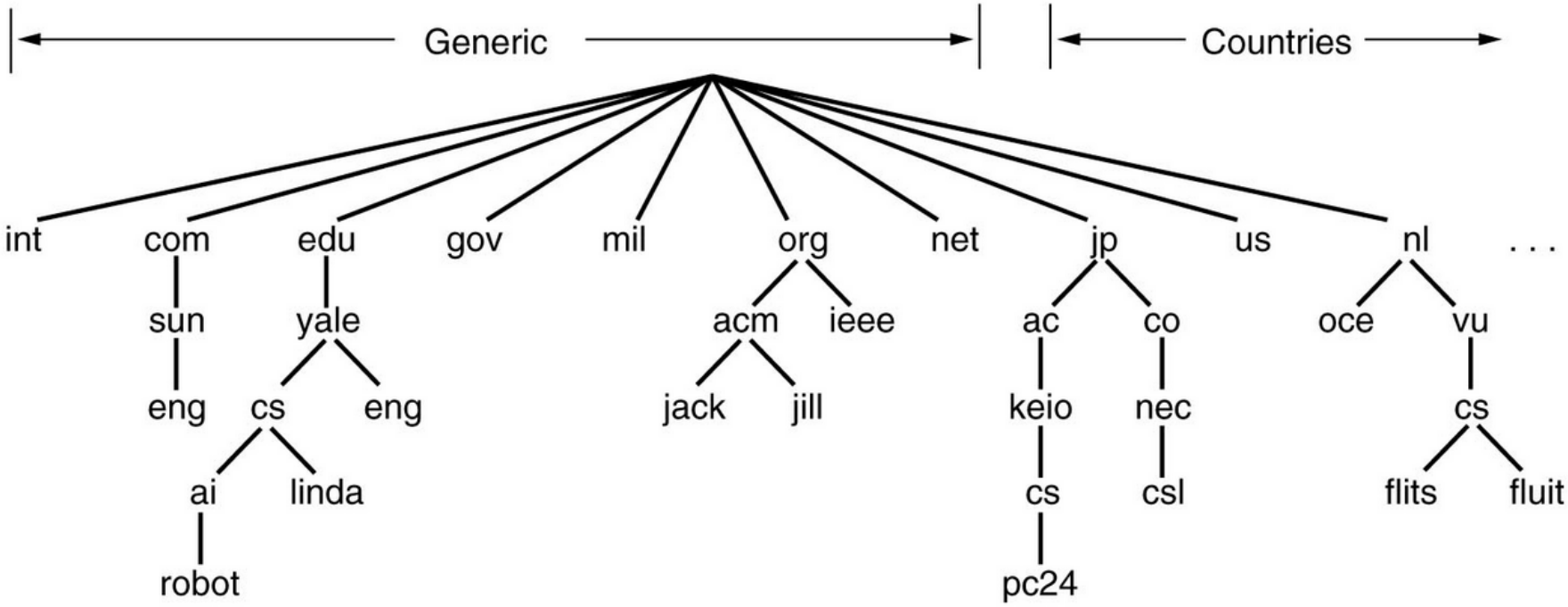
Domain Names

- Here's an example of the hierarchy of names in a Domain Name.
- Below the conceptual "Root" is all of the "top level" names. Generic top level names are ones such as com, net, org, gov, and mil. Country code top level names are ones such as us (USA), ca (Canada), jp (Japan) and nl (Netherlands).



Domain Names

- Below the top level names are the subordinate names. In this example, eng.sun.com is a fully qualified domain name.
- Each label can be up to 63 characters and the fully qualified name cannot exceed 253 characters.
- The name space rarely exceeds five levels.



Name Server

- Domain Names can apply to many different things, but the most common are web hosting and email servers.
  - Different IP addresses may be registered for different functions under the same domain name. Meaning that the web server for example.com may be on a different machine, and a different IP address, than the email server for xxxx@example.com

Finding an IP Address for a Domain Name

- When a client has a domain name, its “name resolver” application will first check its own cache to see if that name was resolved recently.
- If not, it will make a request to the local name server to obtain the IP address.
- The local name server has a cache of recently requested names. If the name is in the cache, it returns the record immediately.
- If not there, conceptually, it will contact the top level name server and work down the tree until it resolves the name.
  - The reality is somewhat different but I won’t go into the details here.

The World Wide Web (WWW)

- The World Wide Web consists of a large set of documents called “web pages”, accessible to users of the Internet.
- The documents are often contain links to other web pages, which causes them to be classified as hypertext documents.
- To support access to these hypertext documents, two things are needed: a web browser and a web server.
- Pages are “written” in a language called HyperText Markup Language (HTML) or, more recently, Extensible HTML (XHTML).
- The HTML is carried in a TCP packet.

The World Wide Web

- Each web page has a Uniform Resource Locater (URL) to identify it. The format of a URL is

  http://www.domainname:port/path/pagename

  - http indicates the protocol. It is followed by a colon and two slashes.
  - WWW is a subdomain of the domain name and was set up to service web requests. Most web browsers will add www if you don't put it in the URL.
  - Domain name is the name of the server. You can also use the actual IP address.
  - For web pages, the port is 80 and can be left off. If the server does not use port 80 you'll have to specify the port number.

The World Wide Web (WWW)

  - Path is the file path to the web page.
  - Pagename is the name of the resource, usually a web page.
- While case (caps or small) is not considered for domain names, case is important in the web page name.
- Domain names are limited in the characters that can be used, while web pages are not. Even the space character can be used, but it must be entered as an % character (%20), or the browser will convert it to a % character.
  - Essentially any name that is valid on a computer is valid as a web page name.

HyperText Transfer Protocol (HTTP)

- HTTP is carried by TCP.
- HTTP is fairly simple, consisting of a limited number of commands – about 9.
- The most commonly used one is “Get” which is sent by the client to the server to request a web page. Others are Head, Post, Put, Delete, Trace, Options, Connect, and Patch.
- Web pages are written in HyperText Markup Language (HTML) and are interpreted by the browser. The HTML instructions cause the page to be formatted for display.

Multiprotocol Label Switching (MPLS)

- We talked earlier about networks with virtual circuits (VC) – especially frame relay and ATM.
- Virtual circuits have certain advantages:
  - Deterministic path through the network, generally with controlled jitter. Important for real time voice and video.
  - Packets arrive in order.
  - Allows for better traffic management.
  - Most VC systems use label switching and label switching is more efficient (faster) than routing.

Label Switching

- Label switching systems establish a route during connection setup. This is done by the ingress router.
- Each router (called a "Label Switching Router -LSR) along the route builds a table which is indexed by the incoming label, and indicates the next hop and the new label.
  - The old label is replaced by the new label before the packet is forwarded to the next hop.
  - This concept offloads the core routers – compute intensive routing done by ingress router.
- Since the table does not have to be searched, as is done in IP routing, switching can be faster. However, new VLSI implementations of IP routing make this a moot point.

So What is MPLS?

- MPLS takes some of the concepts of Frame Relay and ATM and generalizes these ideas to multiple protocols.
- MPLS is an efficient encapsulation mechanism.
- Appends “labels” to data – such as IP packets or ATM AAL5 cells – to switch the data through the network.
- It can be carried over a variety of networks – IP, ATM, etc.
- It can be used to carry a variety of layer 2 protocols – IP, ATM, POS, PPP, Ethernet, etc.
- Most common use is to carry IP packets.

So what is MPLS? (continued)

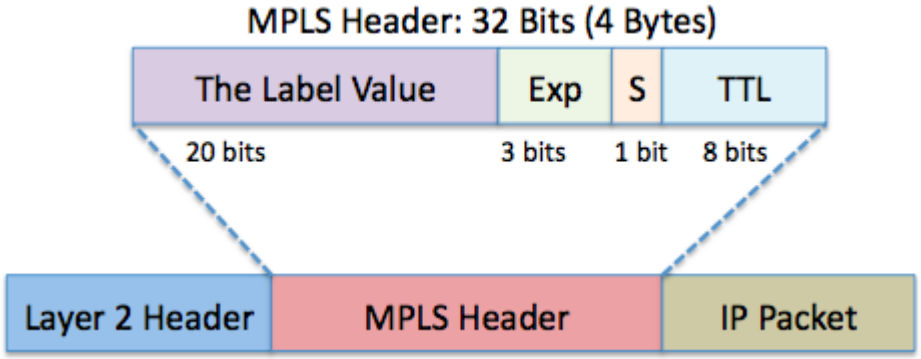
- MPLS adds the concept of a virtual circuit to IP networks.
- When a connection is requested, the first router does a lookup of the final destination router, and then establishes a route to that router.
- Each router in the path makes an entry in its forwarding table for this connection.
- The first router adds a “label” (sometimes called a “shim”) to the IP packet.
- Each router in the path uses the label to determine the next hop and replaces the label (similar to the way ATM works).
- The last router in the path strips the label and passes the IP packet to the destination.

Advantages of MPLS

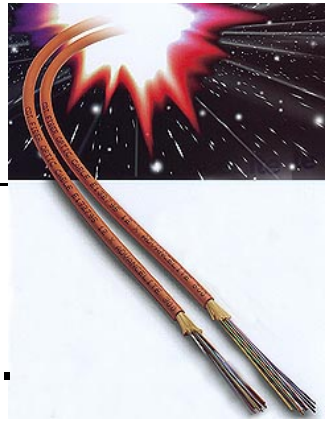
- Traffic engineering: The ability to control where and how traffic is routed on your network, to manage capacity, prioritize different services, and prevent congestion.
- Implement Virtual Private Networks: The ability to provide carriage for a variety of different data services, in addition to IP.
- Improve failure recovery time with the MPLS Fast Reroute.

MPLS Header Format

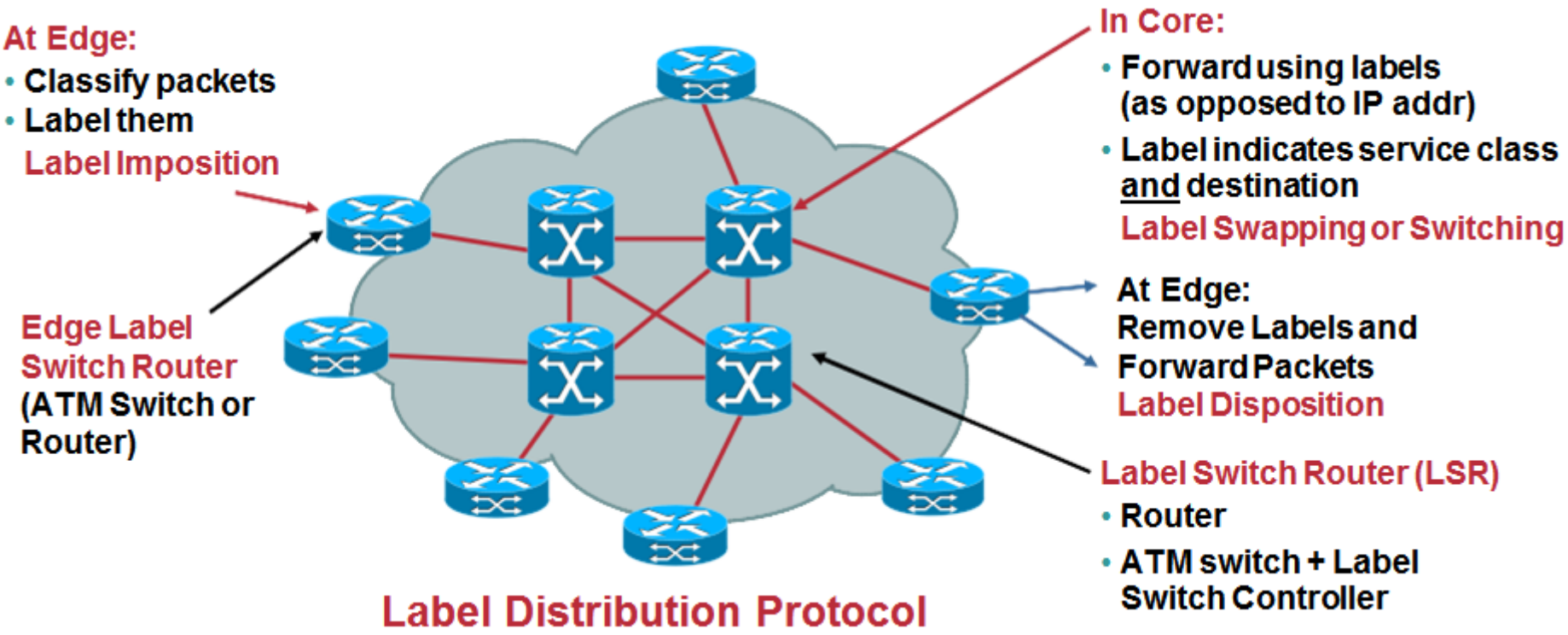
- The MPLS header is 4 octets in length.
  - The Label Value is the path identifier, essentially the same as in ATM.
  - The Exp field is experimental for Quality of Service (QOS) and Explicit Congestion Notification (ECN).
  - S is the stack bit. There can be multiple MPLS headers on a packet. The last header will have this bit set to 1. Otherwise, 0.
  - TTL is Time to Live. The ingress router takes the IP TTL value, decrements it by 1, and places it in the MPLS header.



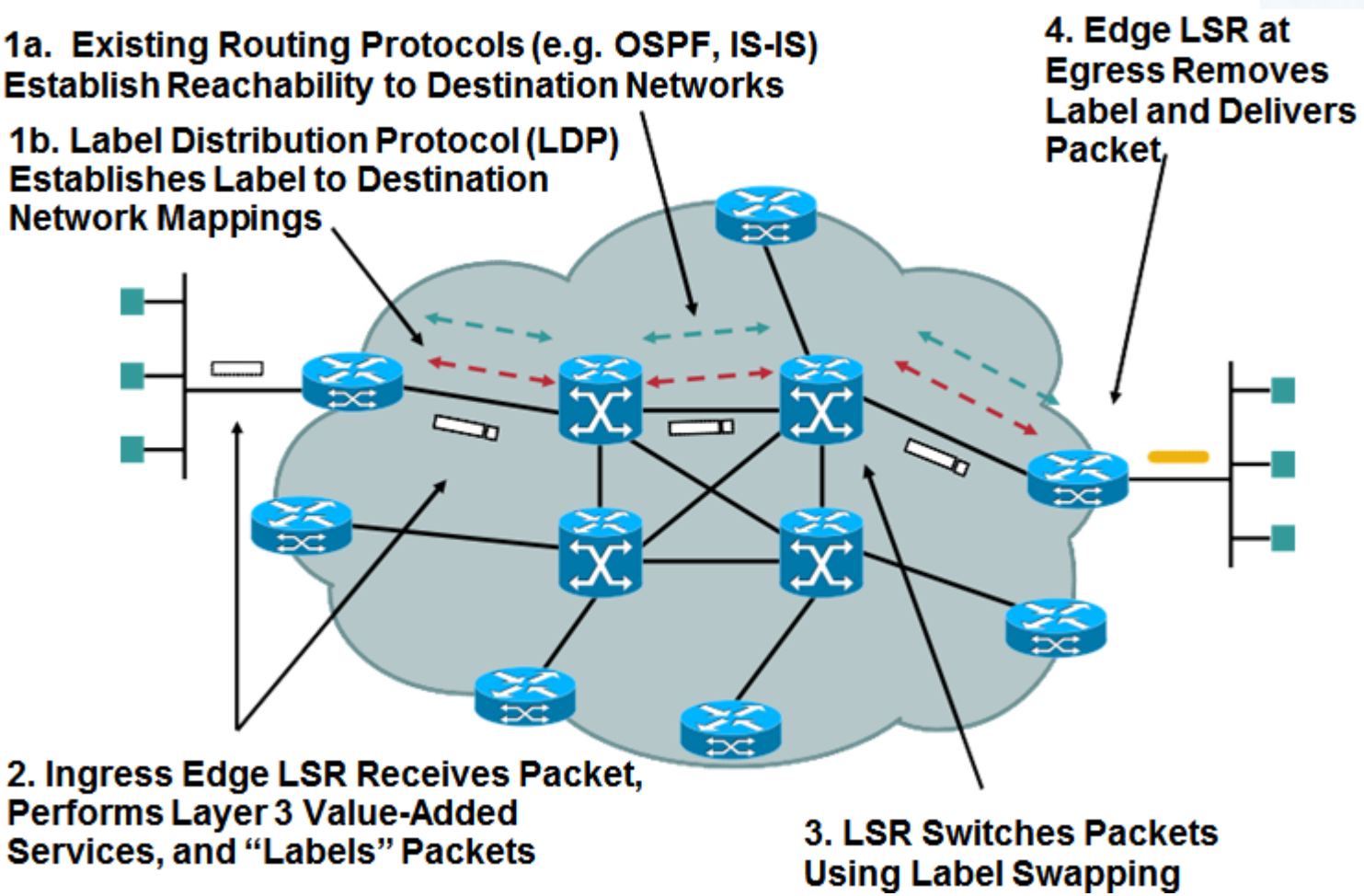
Basic Concepts



- Establish route when the circuit is opened.
- Add labels at the ingress Label Switching Router (LSR).
- Switch through the network using the labels.
- Strip the label at the egress LSR.

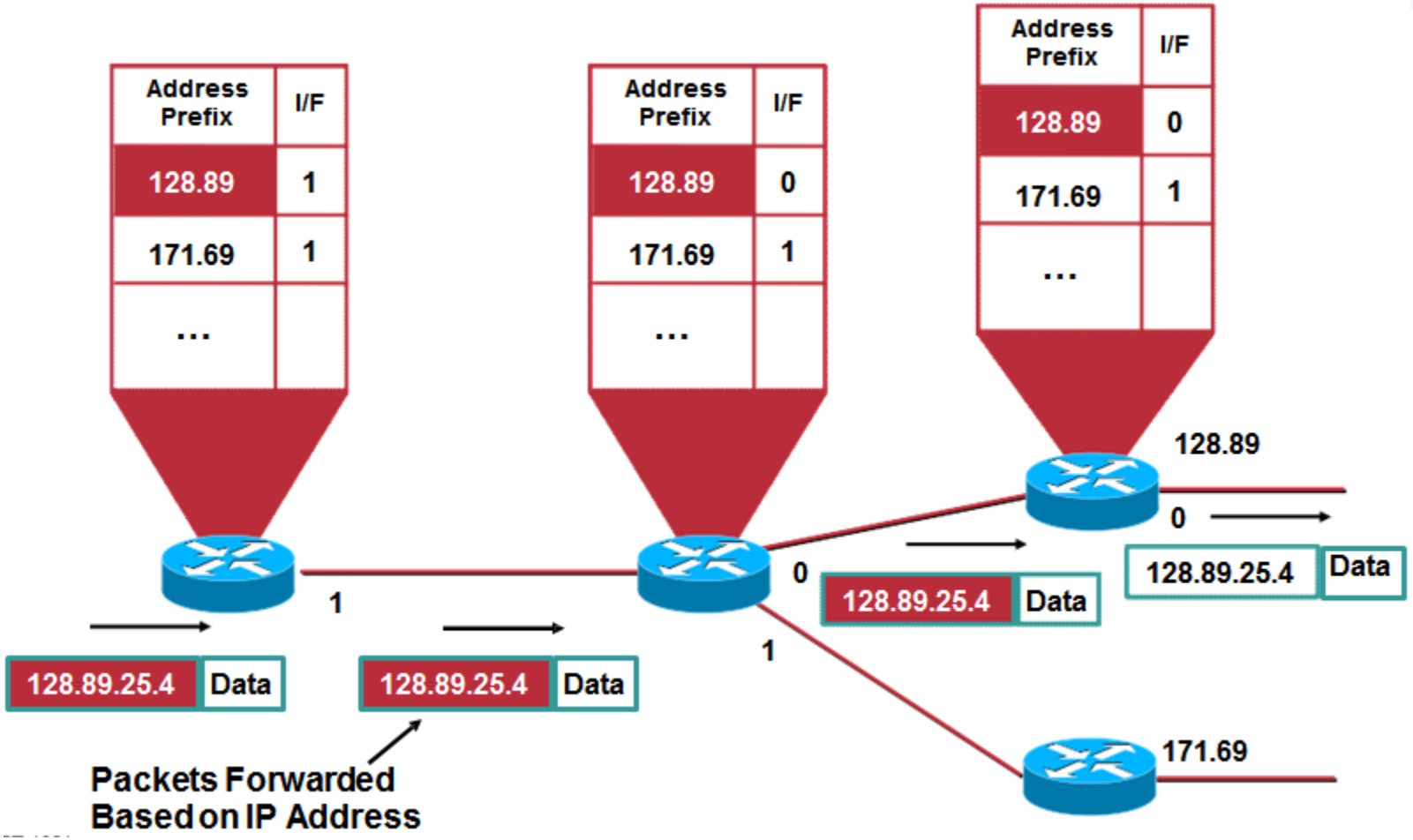


MPLS Connection Setup and Switching



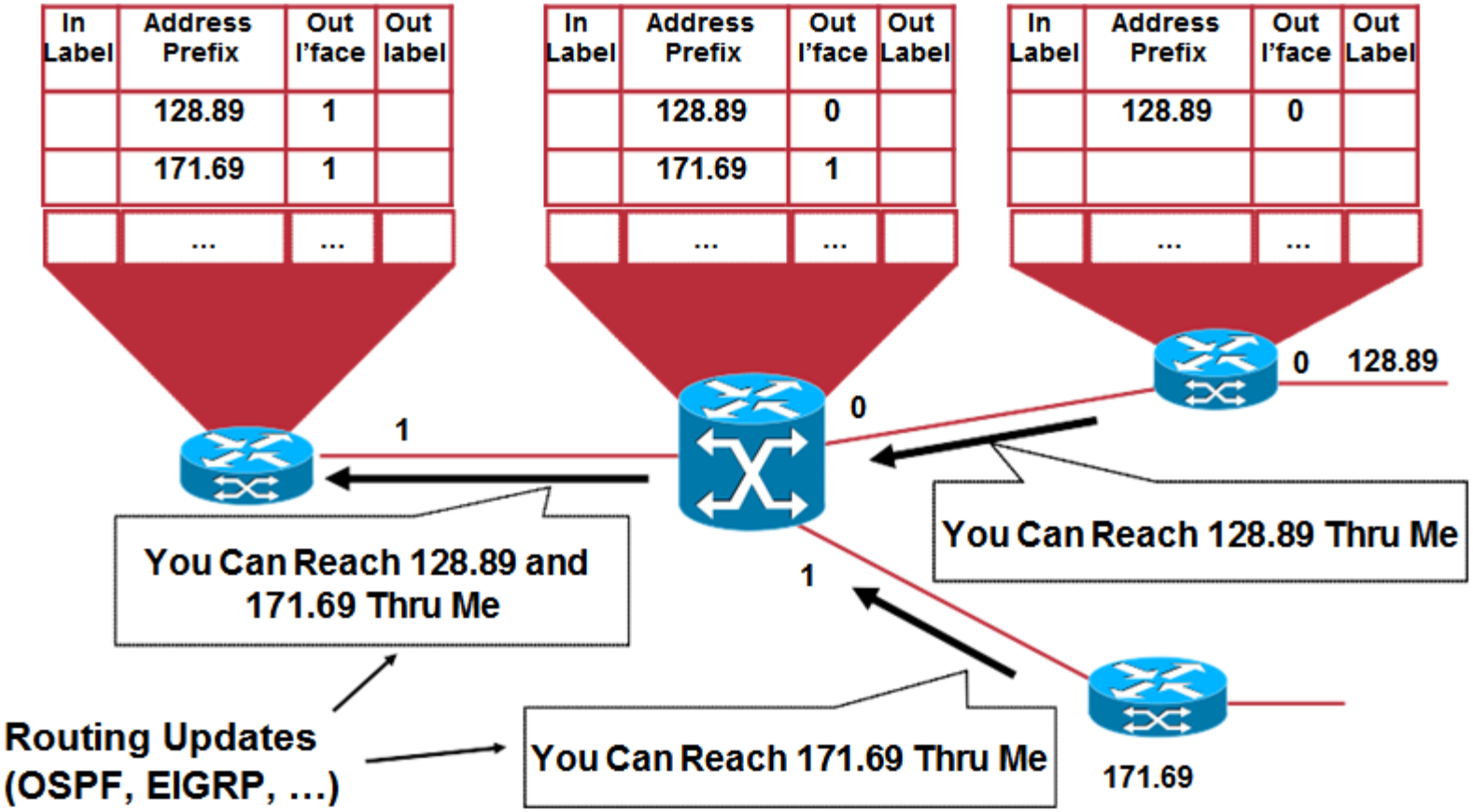
Routing IP

- This is an example of how IP routing works.



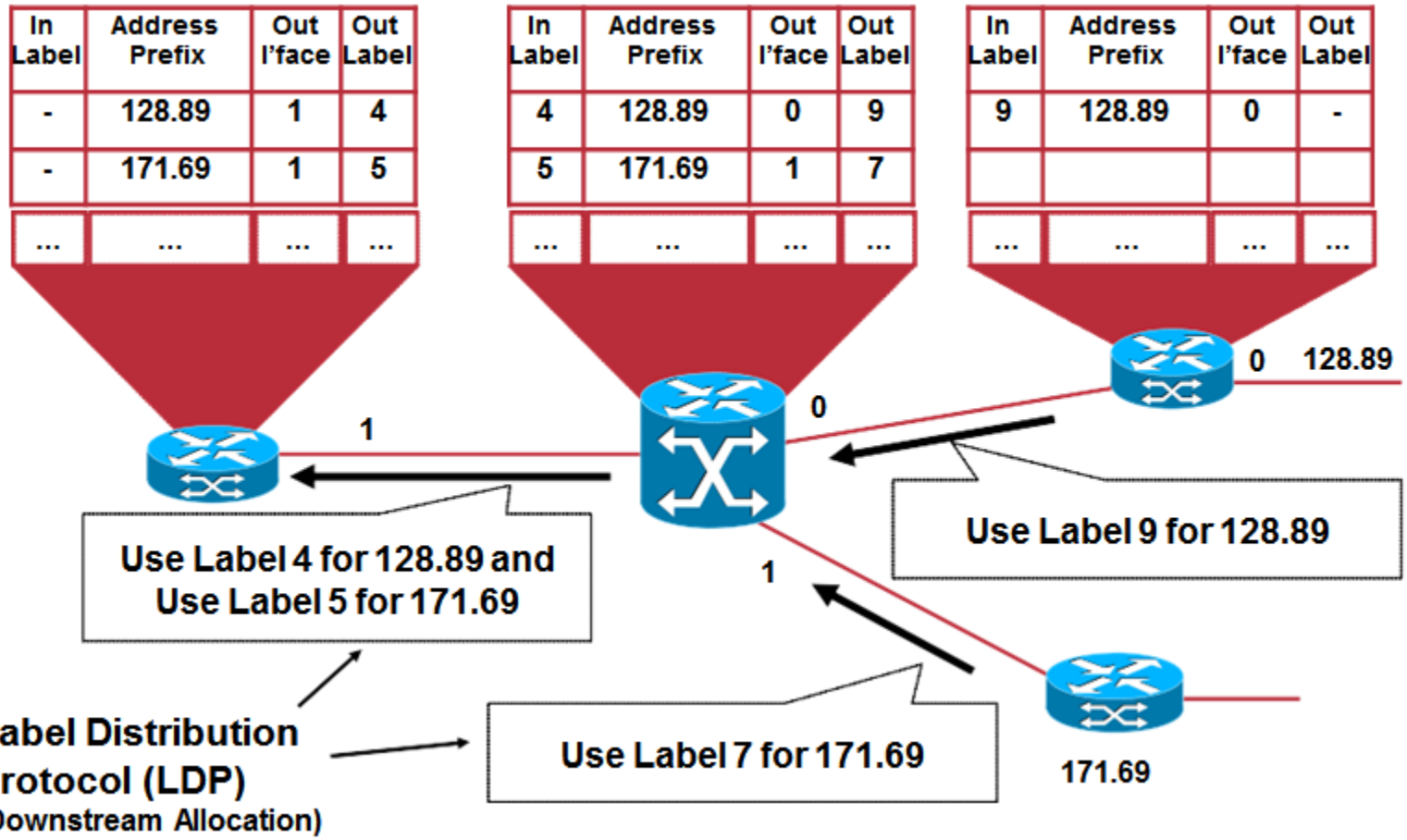
Routing Information Distribution

- Example of standard routing protocol operation.

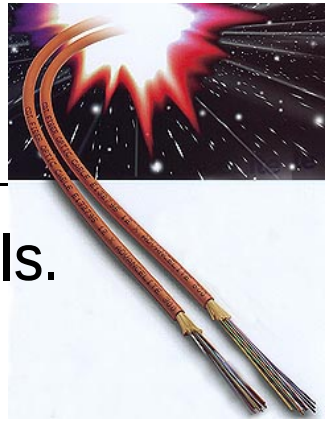


MPLS Label Distribution

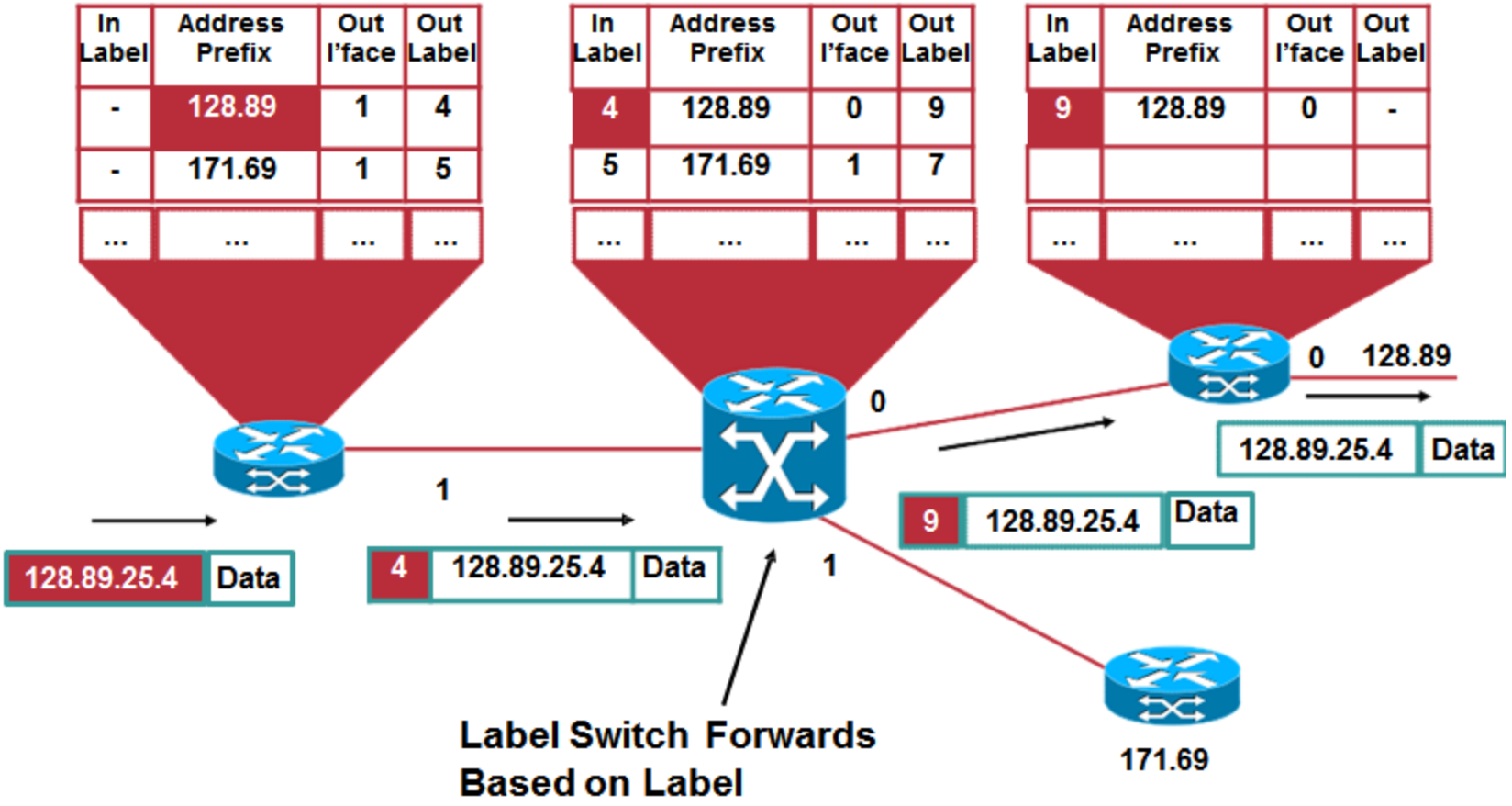
- MPLS Label Distribution adds the allocation and distribution of labels to the “routing” information.



IP Switching based on Labels



- And here's how IP packets are switched using the labels.

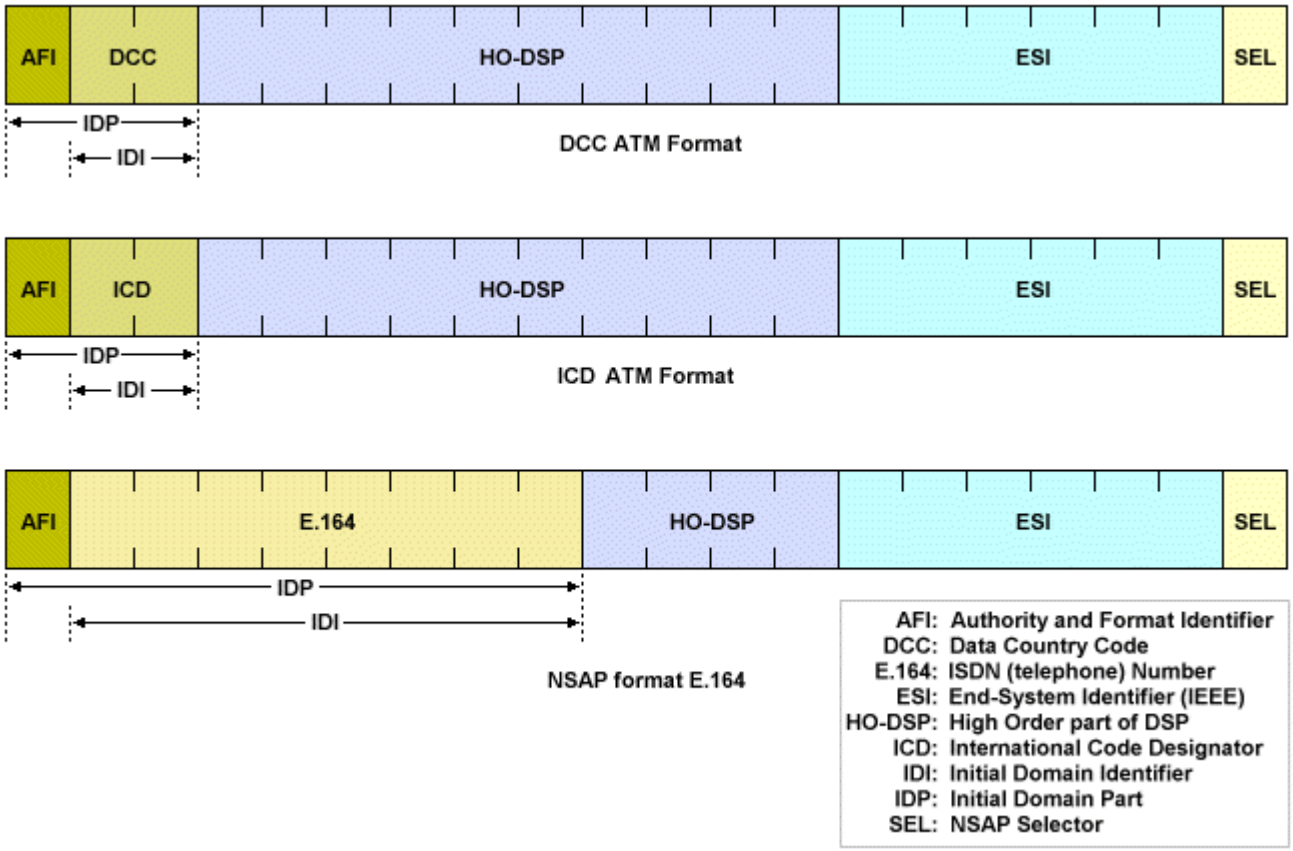


MPLS Summary

- There's a lot more to MPLS than covered here.
- The driving force for MPLS is traffic engineering.
- MPLS can be used to provision Virtual Private Networks (VPN).
- MPLS is well accepted and has been implemented in most large networks, even if you don't see it from the edge.

Questions?

- AESA


AFI
DCC
HO-DSP
ESI
SEL
IDP
IDI
DCC ATM Format
AFI
ICD
HO-DSP
ESI
SEL
IDP
IDI
ICD ATM Format
AFI
E.164
HO-DSP
ESI
SEL
IDP
IDI
NSAP format E.164
AFI: Authority and Format Identifier
DCC: Data Country Code
E.164: ISDN (telephone) Number
ESI: End-System Identifier (IEEE)
HO-DSP: High Order part of DSP
ICD: International Code Designator
IDI: Initial Domain Identifier
IDP: Initial Domain Part
SEL: NSAP Selector