

CS 51 Computer Science Principles

APCSP Module 3: Data, Internet, Computer and Programming

Unit 4: Computer and Networks



LECTURE 6 INTERNET

DR. ERIC CHOU

IEEE SENIOR MEMBER

Objectives

- Transport Layer, Session Layer, Presentation Layer and Application Layer.
- Packet
- Domain Name System
- Distributed Computing
- Network Equipment
- Anti-virus/Anti-spamware



Packets

LECTURE 1

Vocabulary

- **Packets** - Small chunks of information that have been carefully formed from larger chunks of information.
- **TCP** - Transmission Control Protocol - provides reliable, ordered, and error-checked delivery of a stream of packets on the internet. TCP is tightly linked with IP and usually seen as TCP/IP in writing.

Objectives

- YOU WILL develop a protocol for reliably sending a message over an unreliable internet.
- The Internet Simulator has been setup for this lesson to restrict messages to no more than 8 characters each, and messages get dropped with some probability on every hop.
- This is a problem-solving lesson. On the real Internet packet sizes are limited, and transmission is unreliable.
- In this lesson we have set up the Internet simulator to restrict packet-size to be very small - 16-bits for the "to and from" addresses plus only 8 ASCII characters.

Objectives

- Also, the Simulator drops packets pretty regularly. In a set of 10 messages it's very likely one or two will be dropped.
- This is a real problem in the real internet world.

Problem Description

- The problem you have to solve is how to use the 8-ASCII-characters-worth of data to include both a piece of message you're trying to send, as well as information about how many messages (packets) there are in the whole message, and which number this packet is.
- The need for something like the **Transmission Control Protocol TCP**.
- TCP was designed to overcome the inherent unreliability of the Internet.
- A small but non-negligible percentage of packets are lost in transmission because of faults in the infrastructure of the Internet.

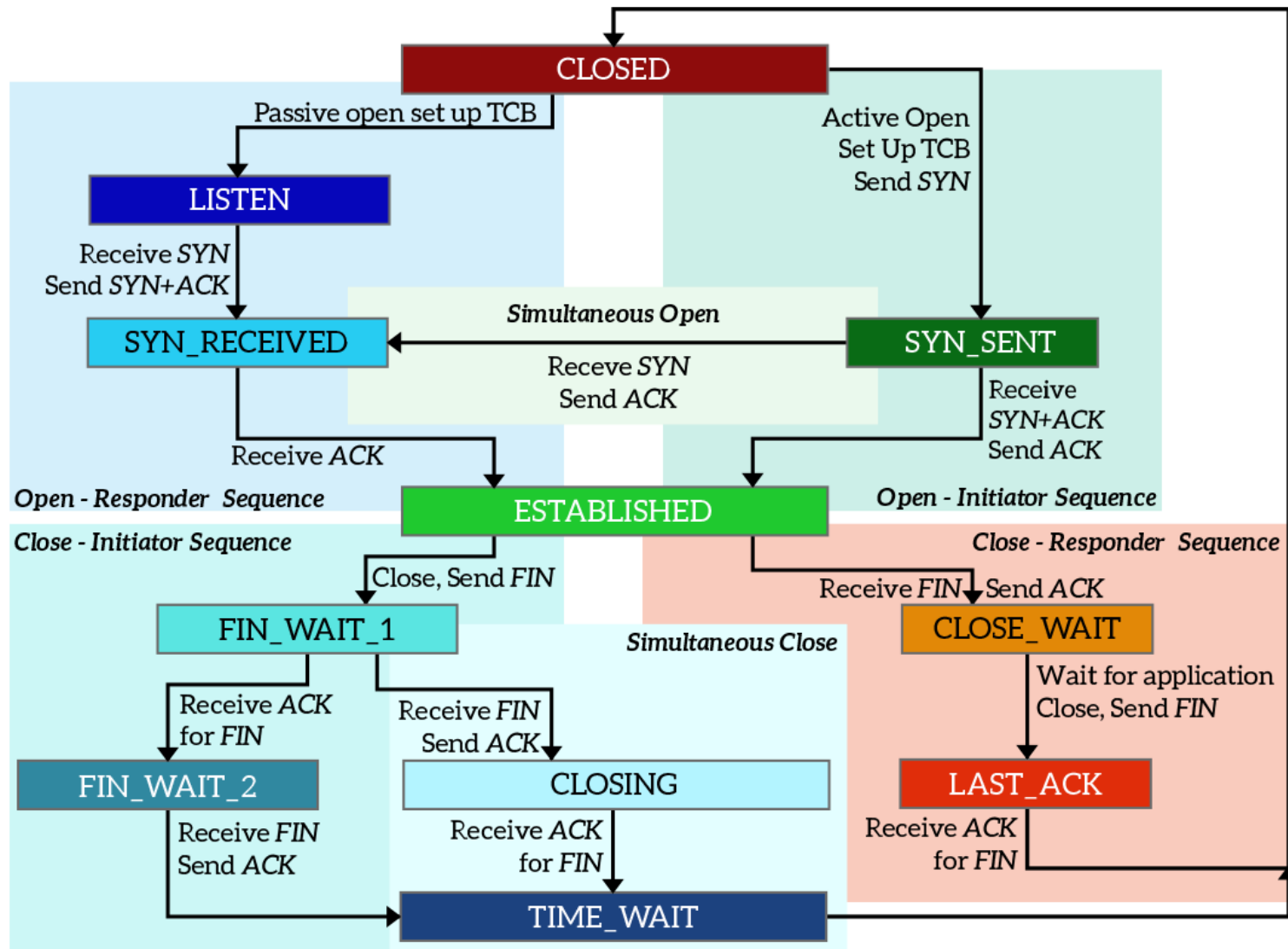
Solution

- To constrain the extent of these errors, larger messages are divided into many packets which are individually routed to their recipient.
- The receiving computer will send an acknowledgement confirming the receipt of that packet.

TCP

- If the sending computer does not receive an acknowledgement, it will resend the packet until all packets have been acknowledged.
- Since packets may arrive out of order, additional data must be included to indicate the order in which the packets should be arranged.
- Thus, while individual packets **cannot** be guaranteed to arrive, eventually an entire message can be accurately reconstructed.

TCP State Diagram



Activity: Internet Simulator

- Your friend sent you a message on the Internet, but you never received it.
- Based on what you already know about routers and the physical Internet, list what reasons might explain this fact.

Problems can be

- Wires are cut
- Interference on a radio channel
- Router malfunctions or cannot keep up with traffic being directed to it

Activity

- When we communicate on the Internet, we are not just sending short text messages as we did yesterday.
- We also use the Internet to exchange documents, videos, music, and scientific data, and these files can easily grow to enormous size.

Solution: Packets

- All of this would not be a problem if the Internet were perfectly reliable, but in reality, errors sometimes occur.
- Wires can be cut, routers can be overwhelmed with traffic, and interference with electric or radio signals can cause messages to become corrupted.
- The response to this problem is to split large messages into smaller pieces of information called **packets**.

Solution: Packets

- It turns out that splitting up a message into packets provides many benefits. If a faster route opens up halfway through transmitting a large file, it is easy to reroute later packets in the transmission through that route.
- Splitting up a message into smaller chunks doesn't solve all the problems of unreliability on the Internet. Packets can still be dropped or arrive out of order.

Activity

- Today's challenge is to develop a protocol to reliably send messages even though the network itself is unreliable.
- The NEW version of the Internet Simulator we will be using today has been structured to simulate the unreliability of the Internet.

Activity

- In particular you'll notice a few changes:
You will only be allowed to send packets containing 8 characters of text!
- Anything larger than 8 characters will be cut off...
However, you may construct multiple packets prior to sending them, by clicking "Add Packet", and then send them all with one click of the "Send" button.
- Every message has a small chance of being dropped on each “hop” it makes between routers.

Activity Guide

Packets and Making a Reliable Internet

- GET in groups of 2 to 4 members.
Students (individually or as a group, either is OK)
- Should log into Code Studio and access the Internet Simulator.

Internet Simulator Setup

- We need to Add at least four routers (six or more is ideal; there's no need to max out the connections on each router)
- It's fine if only one or two students are connected to a router)
- Join a DIFFERENT router from your groupmates.

Internet Simulator Setup

- Take some time to get used to the changes introduced to the **Internet Simulator**.
- Exchange messages across routers.
- Try to construct a multi-packet message and send multiple packets at once to someone else try the same thing for a classmate on ANOTHER router and then view the router logs to examine the result of these transmissions.

Activity

- Send a drawing with ASCII text and come up with ways to break it up.
- The wider the drawing is, the more challenging the protocol will be to develop since each individual message is limited to only 8 characters.

```
#####          v
@   @   .-., |
##### ((.))/
@   @   / \
##### /__\
      _/  \_
```

ASCII ART IS FUN

Discussion

You should recognize:

- Packets are dropped with some frequency
- Packets of more than 8 characters are always truncated to just the first 8 characters
- Packets sometimes arrive out of order.

Discussion

- Develop a protocol that will allow you to overcome the unreliability of the network so that a message can be sent and both sender and receiver can be confident the full message was received.

Guidelines for Protocol:

- All communication can only be done through the Internet Simulator.
- The full message sent will be at least 80 characters long - broken into at least 10 packets - and might be entirely random (i.e. there's no way to use human intuition to reconstruct the message).
- The message is not known beforehand.
- The sender and receiver must be confident the full message was successfully transmitted and reconstructed.

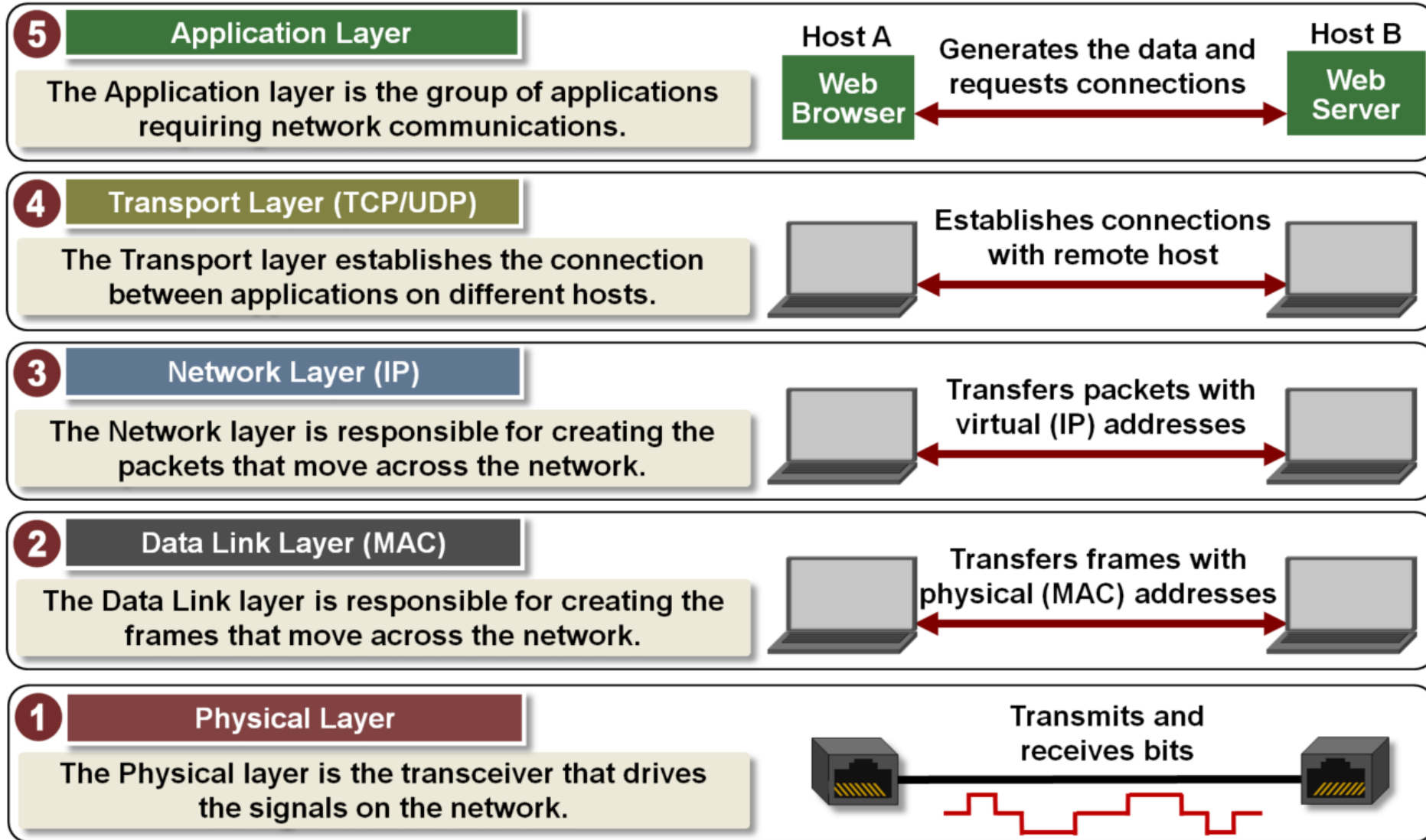
Guidelines for Protocol:

- The real problem to solve is to think about what the recipient of the messages should do to inform the sender of what's missing and needs to be re-sent.
- It's tricky because any message the recipient sends back to the sender also stands a chance of being dropped or lost.
- Your protocol needs to **overcome this unreliability** for both the sender or receiver.

Transmission Control Protocol (TCP) Header

20-60 bytes

source port number 2 bytes				destination port number 2 bytes			
sequence number 4 bytes							
acknowledgement number 4 bytes							
data offset 4 bits	reserved 3 bits			control flags 9 bits			window size 2 bytes
checksum 2 bytes				urgent pointer 2 bytes			
optional data 0-40 bytes							



Video

The Internet: Packets, Routing, and Reliability

(about 6 minutes)

Connect activity to TCP

- The challenges we encountered in today's activity very closely mirror those that exist on the actual Internet.
- The response was the development of a protocol called the Transmission Control Protocol, or more simply, TCP.
- TCP divides larger messages into **smaller** packets which have ordering information added to their header.
- When a packet arrives at a destination computer, an acknowledgement is sent to the sender, letting them know they don't need to resend that packet.
- Once all the packets have arrived, the ordering information in the headers of the packets allows them to be reordered to create the original message.

Practice

DO your reflections and assessments in **code studio**.



Parallel and Distributed Computing

LECTURE 2



DNS

LESSON 6 [UNIT 2 CODE.ORG]

Vocabulary

- **DNS** - DOMAIN NAME SYSTEM - The service that translates URLs to IP addresses.

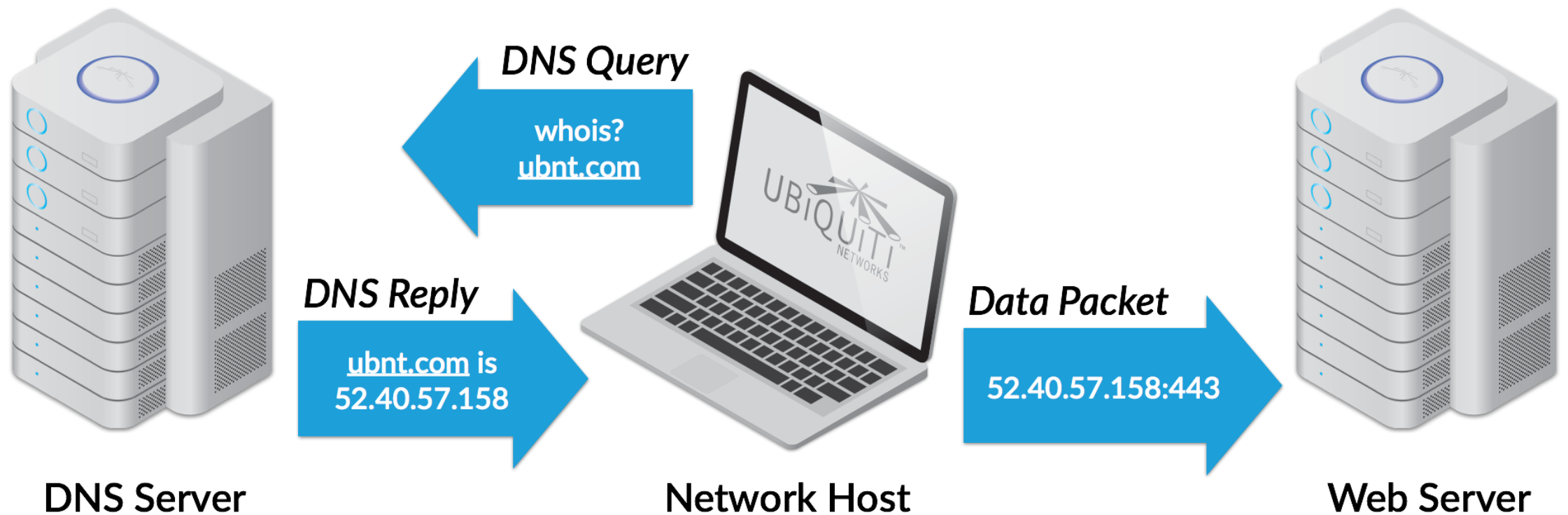
Lesson Materials

1. One (only one) IP Address Label
2. Worksheet for names and address
3. Activity Guide – DNS partner questionnaire
4. Activity Guide –Research: DNS in the Real World
5. Homework for Lesson 12

Domain Name Server

- The core idea of this lesson occurs in the unplugged activity that kicks off the lesson, in which students try to keep track of IP addresses that had been randomly assigned to each student in the class, while at the same time the students' IP address changes.
- This leads to identifying the need for an authoritative system for name-to-address mappings, known as the Domain Name System or DNS.

DNS Lookup & Internet Traffic



Domain Name Server

- You will then briefly experiment with a DNS protocol in the Internet Simulator. The activity is similar, in that you will have to grapple with IP addresses changing in real time and use the built in DNS protocol to resolve the issues.

Denial of Service Attacks

- The lesson ends with you doing some rapid research about DNS and some of its vulnerabilities, particularly what are known as **Denial of Service Attacks**.

Purpose

- The basic purpose of this lesson is to show some of the challenges solved, and created by, **DNS**. At its core, the DNS is "simply" a hierarchical system of computers and databases, that maps IP addresses to domain names.
- It enables Internet users to connect human-language locations on the Internet with numeric addresses used by **IP**.
- While distributed and hierarchical, it can be treated in the abstract as a centralized registry of locations on the Internet, allowing users to quickly find locations they are looking for and register themselves so that others may find them.

Activity

- USE THE IP ADDRESSES and THE WORKSHEET.
- For the next 5 minutes, your goal is to complete an accurate list of IP addresses and names for all students in the room. You may only talk to one person at a time, but you may exchange as much information with that person as you want.

IMPORTANT: From time to time, get a NEW IP Address from the collection of addresses.
GO!

Domain Name and IP Addresses

- Why did we switch your IP addresses?
- This simulates the fact that a computer's IP address does not stay the same.
- For example, a person's IP address on their phone changes quite frequently as they move around throughout their day and their phone tries to connect to the Internet from different locations.

Domain Name and IP Addresses

- Do you think the system we just simulated is an efficient way of collecting IP addresses? Are there any inefficiencies you observe? How could it be made better?
- A central list would be better, and the **Internet** has a system for that.

Activity Guide

- Open Code Studio and connect to the Internet Simulator and use the **DNS Partner Questionnaire - Activity Guide**.
- The new configuration of the simulator includes a DNS server.
- A DNS server now appears attached to every router.

DNS Protocol in Experiment

- We no longer can see anybody's IP address. To get an IP address, we have “ask” the DNS server using a text-based protocol.
- Try this: see how to send a request to the **DNS** for someone's address.
- Try the **DNS** protocol to get the address of someone who is attached to their router.

DNS in the Internet Simulator

When you go to the internet simulator now. You will see a “DNS server” attached to the router. In order to communicate with someone else, you must first find their IP address by asking the **DNS**.

1. To begin, click over to the “DNS” tab to see all the hostnames of people on the router. You will see the address of the DNS (always 15) but will not see an address for anyone else on the router.
2. The DNS server responds to a text protocol that will give you someone’s IP address. The protocol is:

GET <hostnameOfPerson>

For example,

GET madeline4


DNS

- After the DNS has returned an IP address, you can type that IP address into the “To” field, enter a message, and then press “send.”
- You are going to interview/have a conversation with a classmate using only the Internet Simulator. We’ve created a list of interview questions (on the back) and you should both jot down each other’s responses.
- To find the person, you will have to ask the DNS for her IP address. When you have retrieved the IP address, start the interview.

Domain Name Switch

- HOWEVER....As you're working, you both MUST disconnect and reconnect from the simulation. This is to simulate changing IP addresses.
- Do this at least twice during the interview.
- Even though your IP address will change, your hostname will stay the same, so you'll need to re-join a router and ask the DNS for your partner's new IP address in order to continue having your conversation!

ChrisPC DNS Switch Settings



ChrisPC DNS Switch

DNS Settings
DNS Database
Program Settings
About DNS Switch
Visit our Website
Like us on Facebook

DNS Database
Show: All DNS Presets

No	DNS Preset Name	Preferred DNS	Alternative DNS	Category
1	OpenDNS Home	208.67.222.222	208.67.220.220	Regular DNS
2	Google Public DNS	8.8.8.8	8.8.4.4	Regular DNS
3	Comodo SecureDNS 2.0	8.26.56.26	8.20.247.20	Secure DNS
4	Norton ConnectSafe Basic	199.85.126.10	199.85.127.10	Regular DNS
5	Norton ConnectSafe Family	199.85.126.30	199.85.127.30	Family Safe ...
6	Norton ConnectSafe Secure	199.85.126.20	199.85.127.20	Secure DNS
7	OpenNIC Europe	151.236.6.156	5.9.234.2	Secure DNS
8	Swiss Privacy Foundation	77.109.138.45	77.109.139.29	Anonymous...
9	NeuStar DNS Advantage	156.154.70.1	156.154.71.1	Secure DNS
10	OpenDNS Family Shield	208.67.222.123	208.67.220.123	Family Safe ...

Add DNS Preset
Edit DNS Preset
Delete DNS Preset

Protect your privacy & surf anonymously with ChrisPC Anonymous Proxy Pro. Click for details.

ChrisPC DNS Switch 1.00
Developed by Chris P.C. srl

Hide settings

Video

- You may remember (from the IP/DNS video that we saw several lessons ago) that you learned about the Internet system (v DNS) for sharing names and IP addresses. Let's watch that section again!
- Watch the DNS portion of this video as a transition to the next activity.
- Video: **IP and DNS - start at 4:12**

DNS

- Hopefully we all get the basic idea: the **DNS** is the large-scale system that translates human-readable web addresses into their numeric IP addresses so that computers can communicate.
- This system however was not designed to be secure and that has resulted in some major security incidents over time.
- You're now going to learn about some of them and how they work.

Activity Guide

Get in groups of 3-4 people to complete readings.

Use: Research: **DNS in the Real World** - Activity Guide, one copy per student.

Each person in the group picks an article about **DNS** and **DDoS** attacks

- The list of articles can be found on the first student page (bubble 1) on Code Studio for this lesson. **DON'T ALL PICK THE SAME ARTICLE!**
- Complete the first page of the worksheet

Activity Guide

So the front page is about your article that you read.

- Flip the paper over, and share with your group members Key points about their articles.
- Find other people in the room who have different articles, and add those points.

Try to get info on as many articles as you can.

Summary

- The Internet is basically a network of computers sending messages to request information and computers replying to messages to satisfy information requests.
- Computers need to identify “from” and “to” for all messages. Computer speak in numbers, not names.
- All communication online is via IP addresses. However, we are more familiar with human readable names, most notably URLs like “**Code.org**” or “**Google.com**.”
- We need a way to translate **human-readable** names into IP addresses.
- It is inefficient for everyone on the Internet to maintain a table of **IP** addresses.

Summary

- The DNS is NOT centralized, but it's not completely autonomous and distributed like routing, either. There is a hierarchical system of servers to maintain an authoritative table that, like a phone book, others can consult when they need to find an address.
- A properly functioning **DNS** system requires collaborative efforts among all users to ensure it is up to date and accurate.

Practice

- Finish up in the Code Studio with Assessments and Reflections.



HTTP

LESSON 6 [UNIT 2 CODE.ORG]

Vocabulary

- **DNS** - The service that translates URLs to IP addresses.
- **HTTP** - HyperText Transfer Protocol - the protocol used for transmitting web pages over the Internet
- **IP Address** - A number assigned to any item that is connected to the Internet.
- **TCP** - Transmission Control Protocol - provides reliable, ordered, and error-checked delivery of a stream of packets on the internet. TCP is tightly linked with IP and usually seen as TCP/IP in writing.
- **URL** - An easy-to-remember address for calling a web page (like www.code.org).

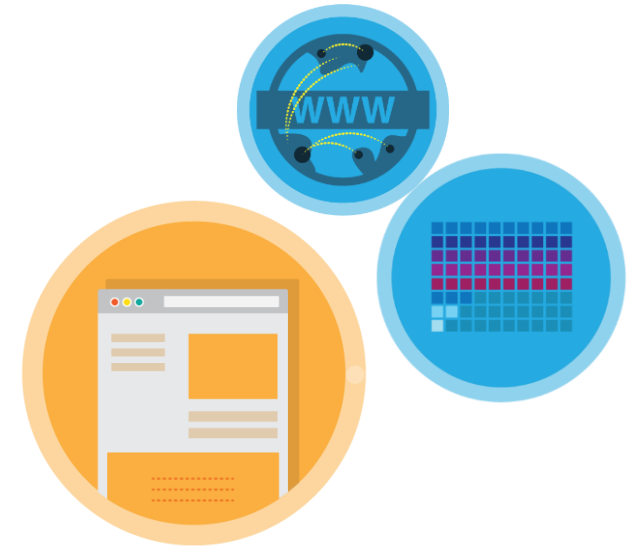
Objectives

- You will be introduced to another high-level protocol of the Internet, **HTTP**. We review the layers of the Internet covered so far, and then watch a video covering high-level protocols of the Internet, most notably **HTTP**.
- You will investigate **HTTP** traffic generated within your own browser by accessing the browser's developer tools and visiting a variety of websites.
- A handout summarizing the structure of **HTTP** is provided to help you understand the components of the **HTTP** requests and responses you will observe. Finally you will see how the layers of the Internet make use of abstraction.

Big Idea

Abstraction

- One of the BIG IDEAS for the College Board is the idea of **Abstraction**



Two Purposes to this lesson:

1. Get a basic understanding of what **HTTP** is and what it's for. **HTTP** like DNS is an ASCII-text based protocol - it's just two computers sending text messages to each other.
 - What makes it a protocol are the rules of the "conversation" the two machines are having.
 - In the case of **HTTP**, it is a **call-and-response protocol** for a client/server relationship, where a client requests a web page or other content (image, sound, video, etc.) from a server. The server looks for it and sends it back.

Second Purpose

2. understand **HTTP** as a "high level" protocol that sits on top of all the other protocols and internet systems we've learned about in the course.

- Text message conversation between the computers is being broken up into **TCP/IP** packets, and all the data gets sent as bits over wires and airwaves, taking different paths, and it gets interpreted reassembled at the end.

Each Internet Layer is a Level of Abstraction

The Internet works in "layers" and this is a perfect example of **abstraction** on the Internet, as one layer makes use of the functionality provided by the layer below it, without worrying about the details of how this functionality is achieved.

HTTP's Functions

- HTTP doesn't have to worry about anything other than the text protocol of how HTTP works. The network software and devices on your and others' computers handle looking up **addresses**, breaking down **data**, **packetizing**, **routing**, **transmission** and **interpretation** and **reassembly**. It's really amazing.

- (1) User issues URL from a browser
http://host:port/path/file



- (5) Browser formats the response
and displays

Client (Browser)

- (2) Browser sends a request message

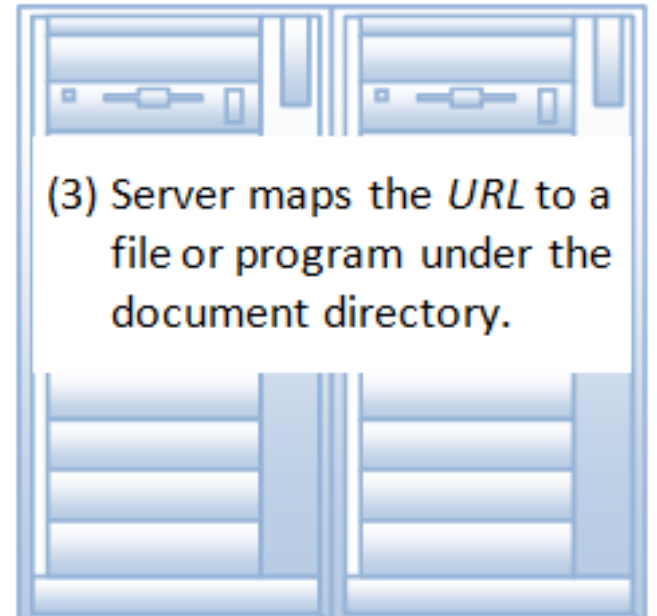
GET *URL* HTTP/1.1
Host: *host:port*
.....
.....

- (4) Server returns a response message

HTTP/1.1 200 OK
.....
.....
.....

HTTP (Over TCP/IP)

- (3) Server maps the *URL* to a
file or program under the
document directory.



Server (@ *host:port*)

How 7-Layer Architecture Works?

- Understand the "Big Picture". You have to be able to recognize that these layers exist and to reasonably explain how they work together for common web page requests.

Layer	Function	Example
Application (7)	Services that are used with end user applications	SMTP,
Presentation (6)	Formats the data so that it can be viewed by the user Encrypt and decrypt	JPG, GIF, HTTPS, SSL, TLS
Session (5)	Establishes/ends connections between two hosts	NetBIOS, PPTP
Transport (4)	Responsible for the transport protocol and error handling	TCP, UDP
Network (3)	Reads the IP address form the packet.	Routers, Layer 3 Switches
Data Link (2)	Reads the MAC address from the data packet	Switches
Physical (1)	Send data on to the physical wire.	Hubs, NICS, Cable

The Internet Protocol Stack

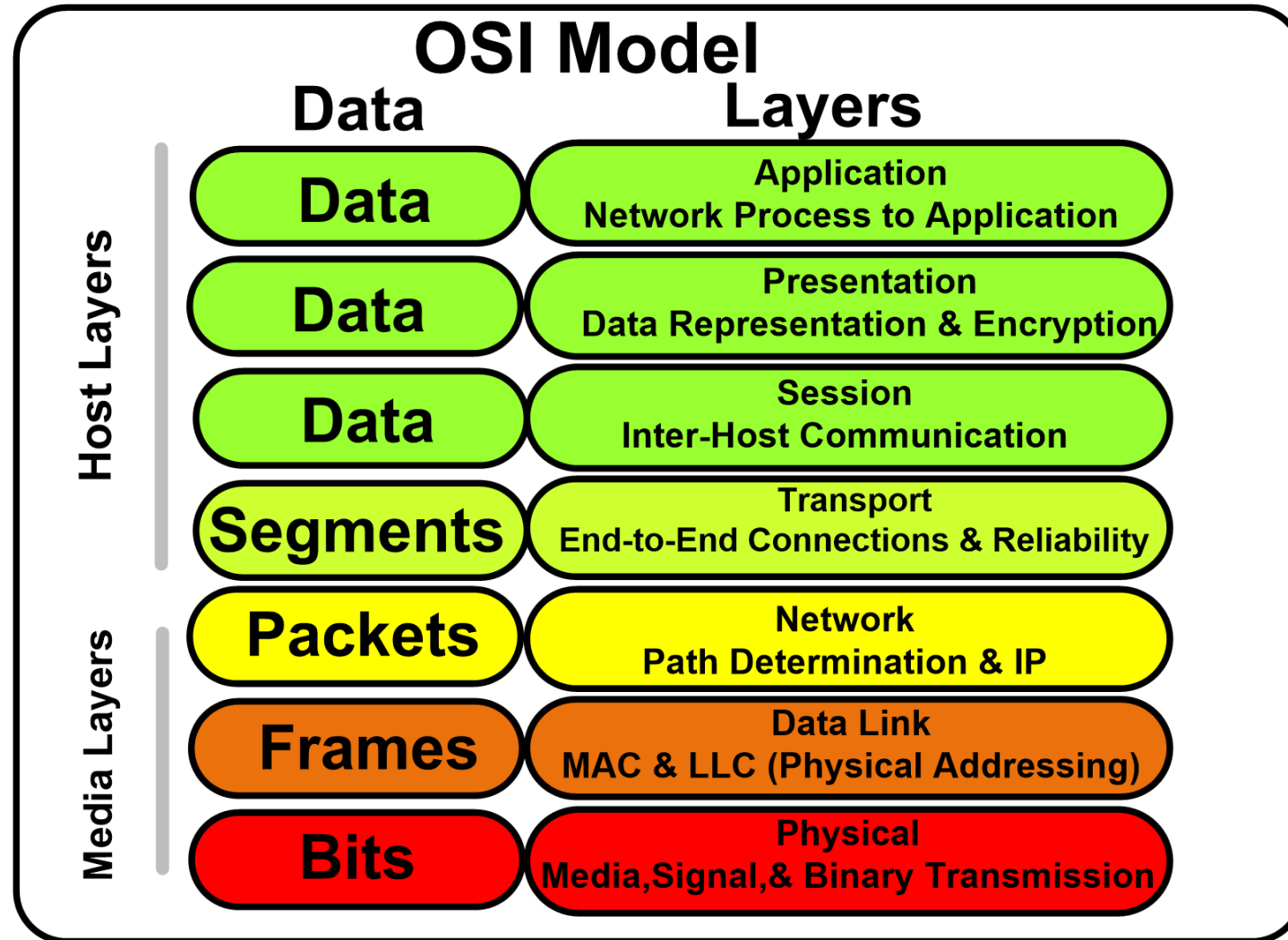
- Make sure you have the “**HTTP and Abstraction on the Internet – Resource**” handout
- We want to think of the protocols as working in “**layers**”.

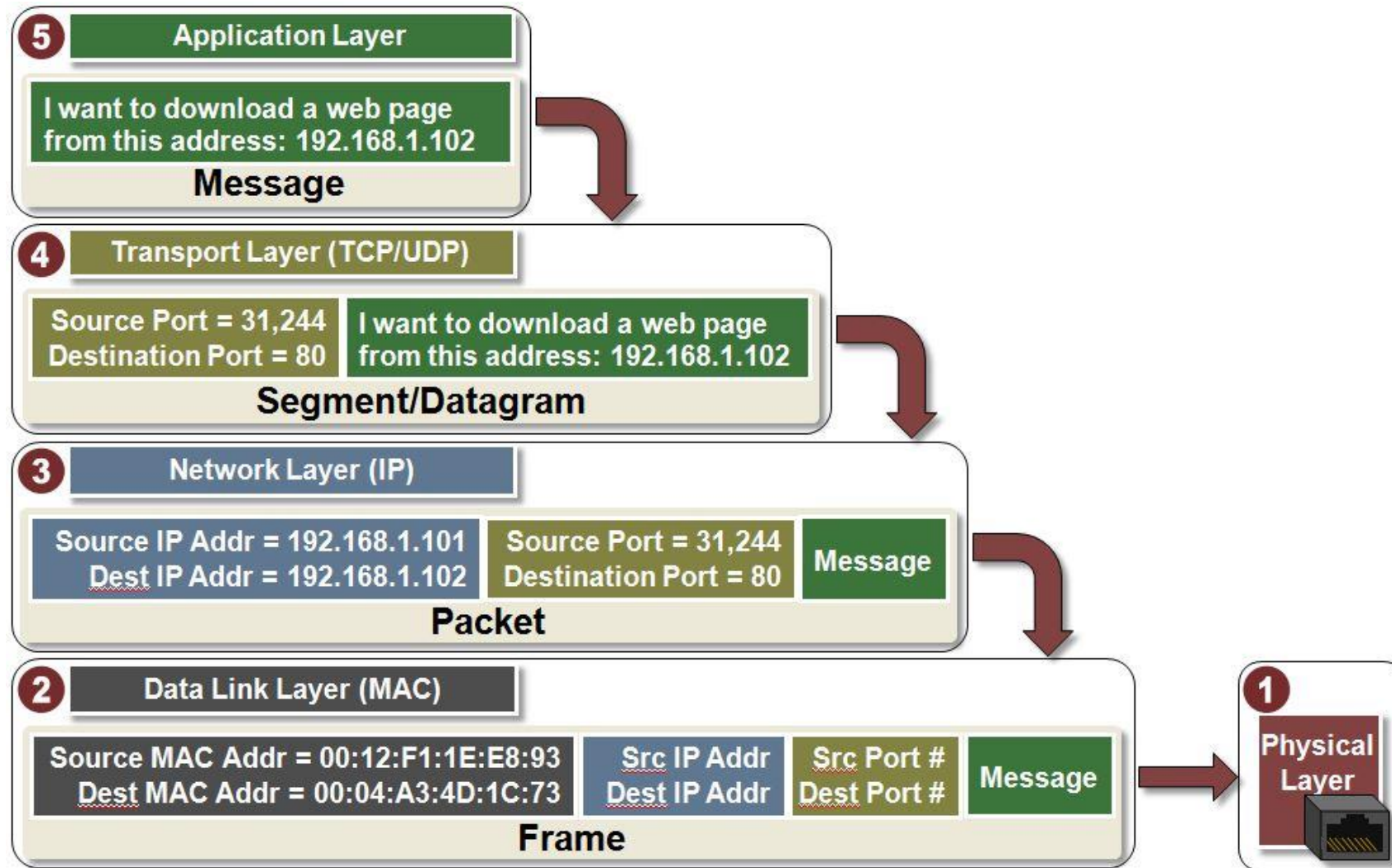
This picture in the handout is a simplified version of what's known as the "Internet Protocol Stack." We've studied each layer separately but now you can begin to see how they work together.

UP/Down Stack

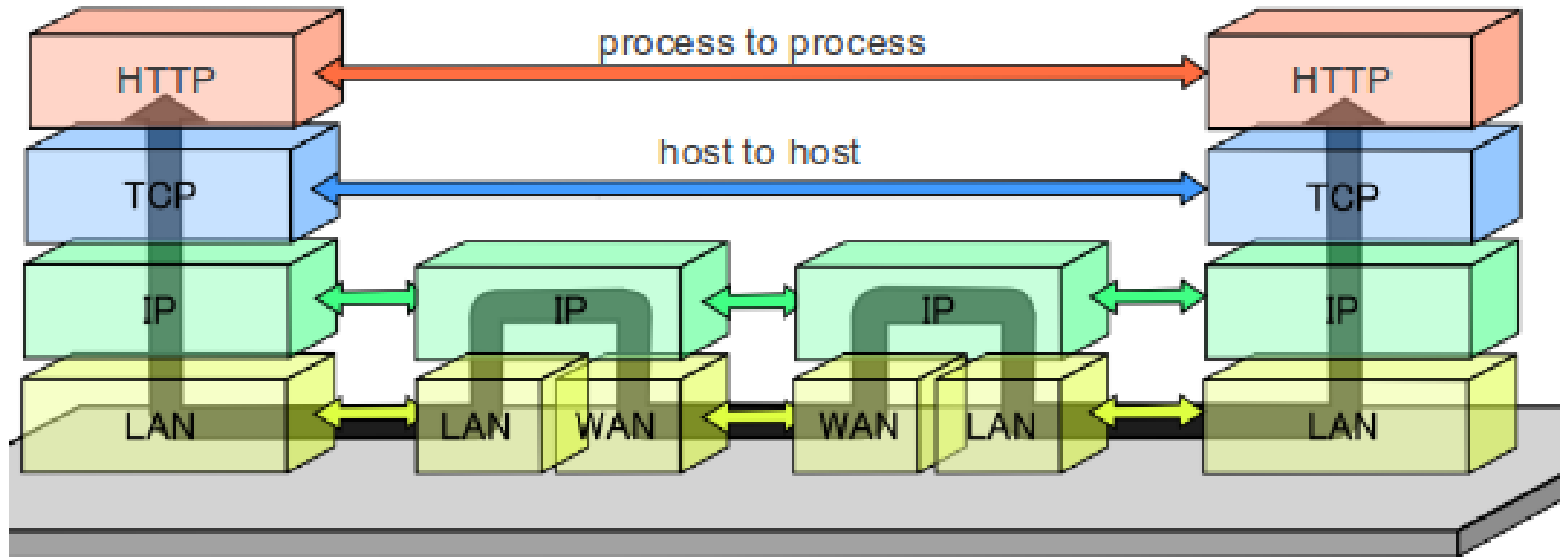
Call and Return of Utility Functions

- We imagine outgoing information going **DOWN** the stack and incoming information going **UP** the stack.
- All of this happens in the **network software** on each computer, whether that computer is your phone, a laptop, or a server like a DNS or web server (routers are a special case that only look at the IP layer).
- The important takeaway here is to understand that the system was constructed with layers of abstraction where each layer only needs to concern itself with its specific job, and then hands it off to another layer. This makes the very complicated task of digital network communication possible.





Data Flow of the Internet Protocol Suite



HTTP

- At the top layer, for example, DNS just thinks is "speaking DNS" to some other computer - the DNS protocol does not even need to know how the other layers work. It just relies on them doing their jobs.
- Have you ever seen the letters "HTTP" anywhere while using the internet?

You see it in the URL of most websites <http://code.org> for example.

- Today we will focus our attention on HTTP, which is a protocol that sits at the same "layer" as **DNS** - right above TCP.

HTTP

- HTTP is an ASCII text based protocol.
- It's somewhat remarkable to note that many "high level" protocols, like HTTP, are just computers sending ASCII text messages back and forth. Each protocol simply defines the rules of the "conversation" between two machines.
- In the case of HTTP it is the protocol used for sending and receiving web pages and other web content.
- Today we'll look under the hood and see HTTP in action.

Video

The Internet: HTML and HTTP

Watch this video in Code Studio.

[about 7 minutes]

Activity

- Get in pairs and use the "HTTP in Action" Worksheet
- Use the other handout as well:
HTTP and Abstraction on the Internet - Resource

Activity

- Access the developer tools of their browser.
- Monitor the HTTP traffic generated by visiting a variety of websites.

Record your findings, using the resource as a guide.

Discussion

- What surprised you about the HTTP traffic you observed?
- What does it mean to say that high-level layers of the Internet use low-level layers “abstractly”?
- What other examples of abstraction have we seen in this course?
- Hint: Unit 1 is basically all about abstraction.

Practice

- Worksheet – HTTP In Action
- Assessment in Code Studio
- Reflection in Code Studio

Homework: Unit 1 – Lesson 13



Network Equipment

LECTURE 3

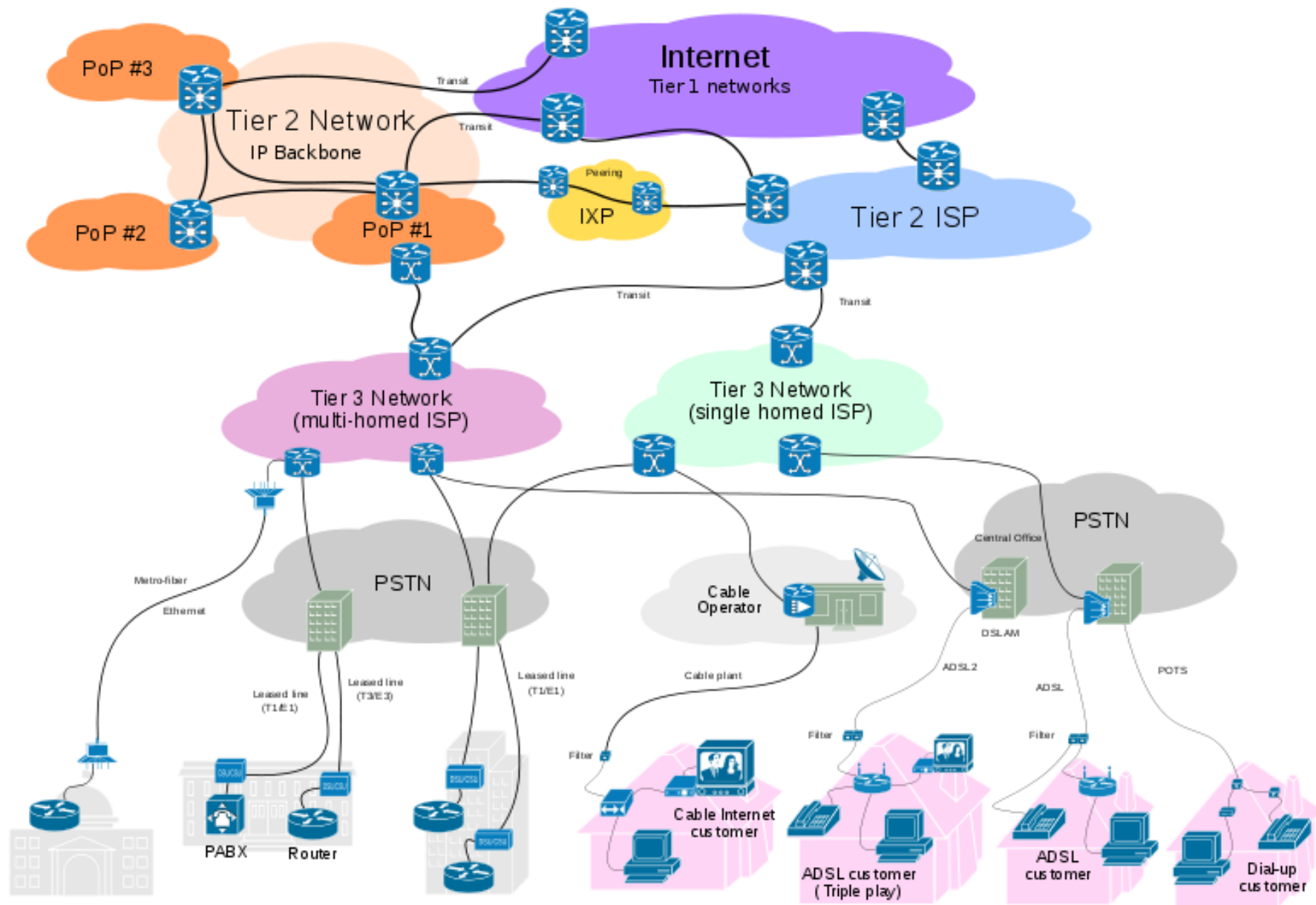
Internet Service Provider

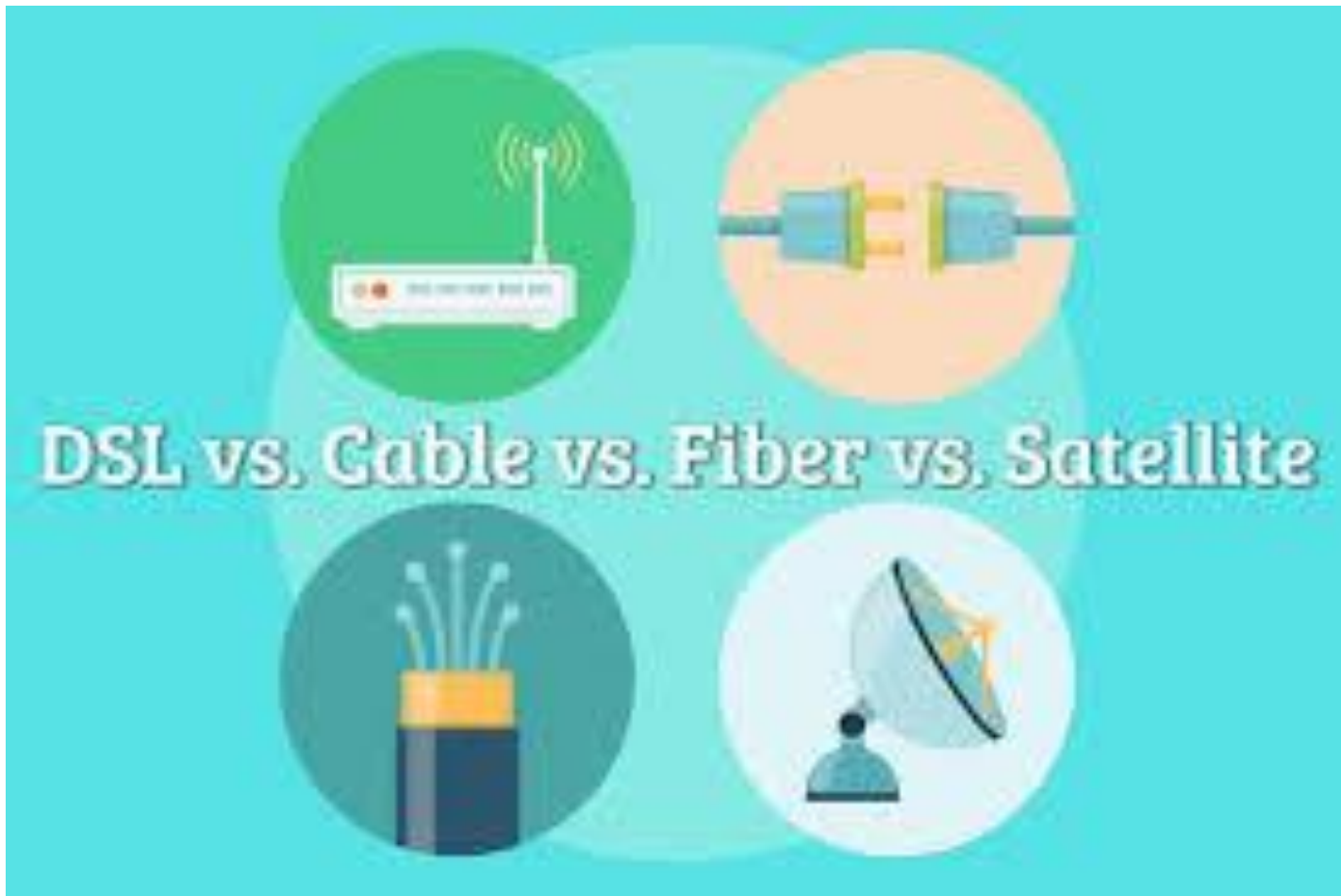
- **An Internet service provider (ISP)** is an organization that provides services for accessing, using, or participating in the Internet. Internet service providers may be organized in various forms, such as commercial, community-owned, non-profit, or otherwise privately owned.
- Internet services typically provided by ISPs include **Internet access, Internet transit, domain name registration, web hosting, Usenet service, and colocation.**



ISP

SECTION 7-1

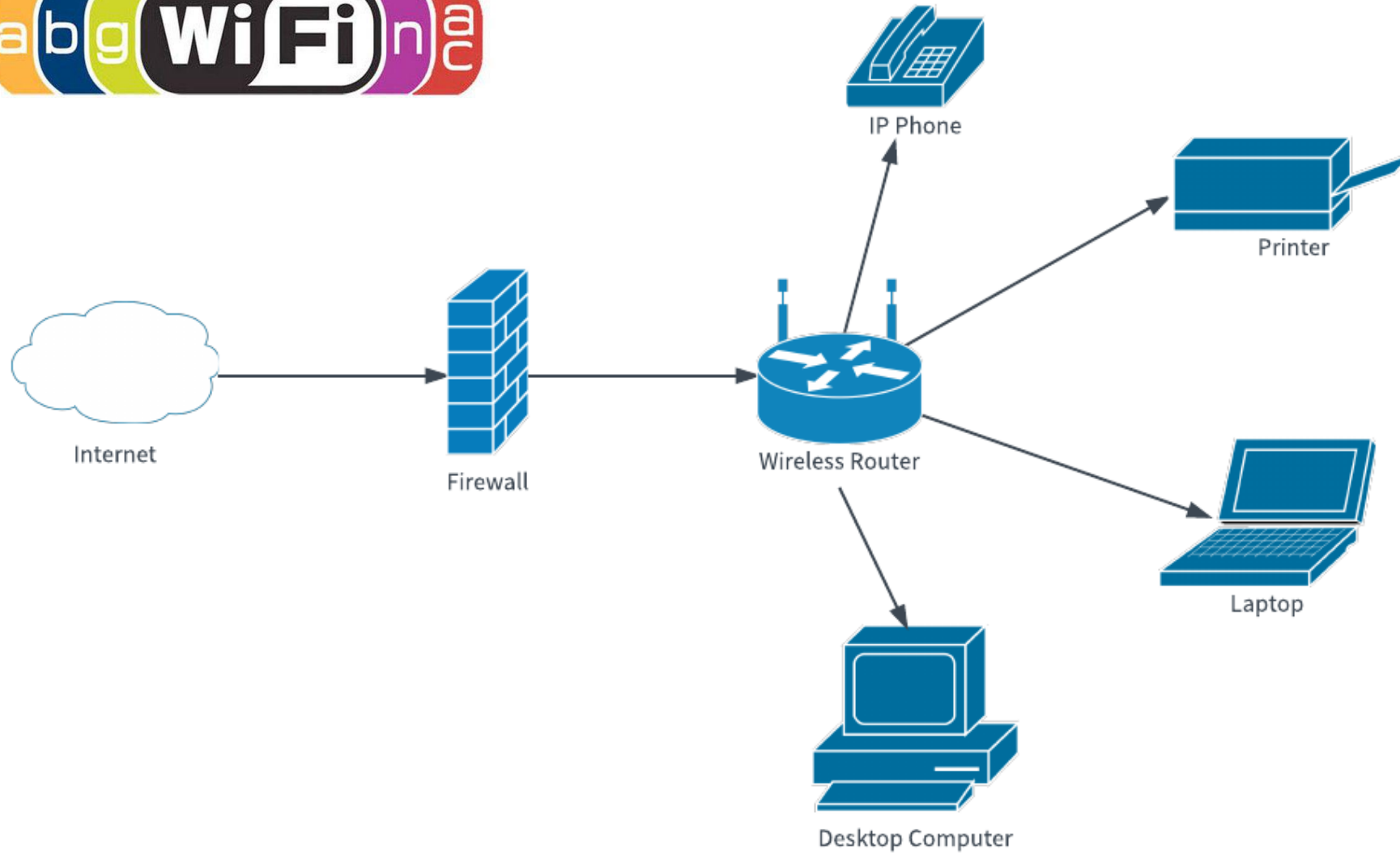




SOHO Routers and Networks

SOHO: Small Office or Home

- A SOHO (Small Office/Home Office) usually consists of a business that is privately owned or an individual who is self-employed, so the term usually refers to both a small office space as well as a small number of employees.
- Since the workload for these types of businesses are often primarily on the internet, they require a [local area network](#) (LAN), which means their network [hardware](#) is structured specifically for that purpose.
- A SOHO network can be a mixed network of wired and [wireless](#) computers just like other local networks. Since these types of networks are meant for businesses, they also tend to include printers and sometimes [voice over IP \(VoIP\)](#) and [fax over IP](#) technology.
- A SOHO router is a model of [broadband router](#) built and marketed for use by such organizations. These are often the same routers used for standard home networking.



SOHO Routers vs. Home Routers

- While **home** networks shifted to predominantly [Wi-Fi](#) configurations years ago, SOHO routers continued to feature wired [Ethernet](#). In fact, many SOHO routers did not support Wi-Fi at all.
- Typical examples of Ethernet SOHO routers were common such as the TP-Link TL-R402M (4-port), TL-R460 (4-port), and TL-R860 (8-port).
- Another common feature of older routers was [ISDN](#) internet support. Small businesses relied on ISDN for internet connectivity as a faster alternative to [dial-up](#) networking.

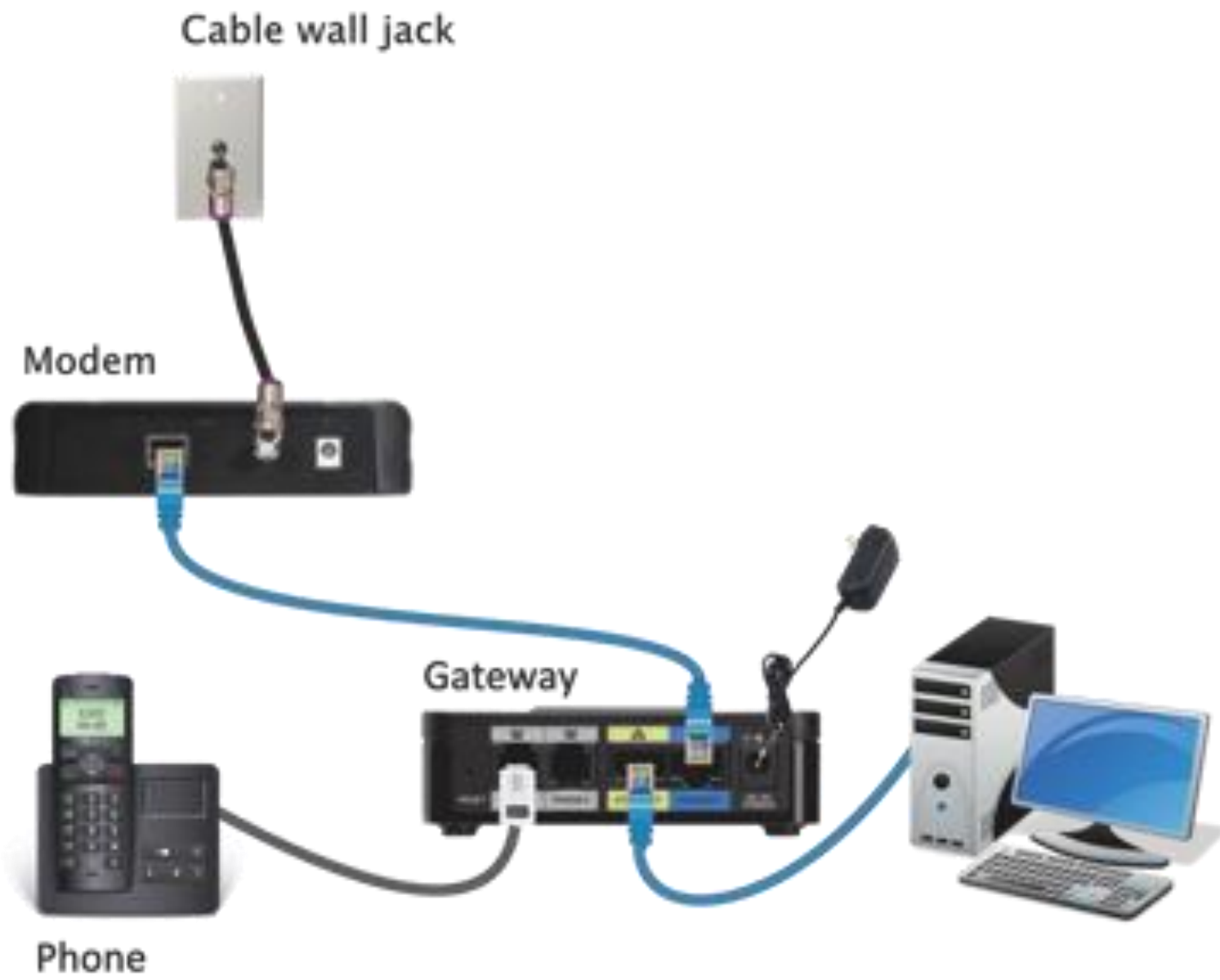


SOHO LAN

SECTION 7-2

SOHO Routers vs. Home Routers

- Modern SOHO routers require most all the same functions as home broadband routers, and in fact, small businesses use the same models. Some vendors also sell routers with more advanced security and manageability features added, like the ZyXEL P-661HNU-Fx Security Gateway, a [DSL](#) broadband router with [SNMP](#) support.
- Another example of a popular SOHO router is the Cisco SOHO 90 Series, which is meant for up to 5 employees and includes firewall protection and VPN encryption.



Other Types of SOHO Network Equipment

- Printers that combine the features of a basic printer with copy, scanning, and fax capability are popular with home office professionals. These so-called all-in-one printers include Wi-Fi support for joining to a home network.
- SOHO networks sometimes also operate an [intranet](#) web, email, and file server. These servers can be high-end PCs with added storage capacity (multi-drive disk arrays).

SOHO LAN Equipments

- WIFI Adapter
- Power Ethernet/Ethernet
- Modem
- Switch
- Router

Side View



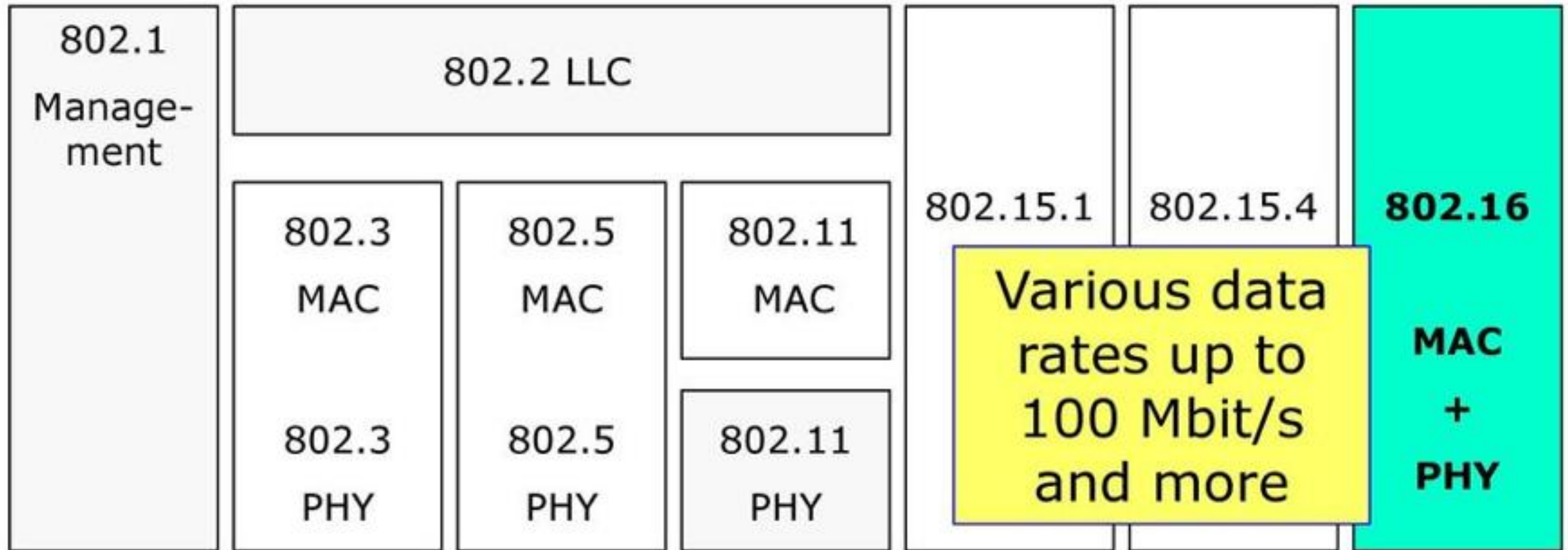


WIFI

SECTION 7-3

WIFI: Wireless Ethernet

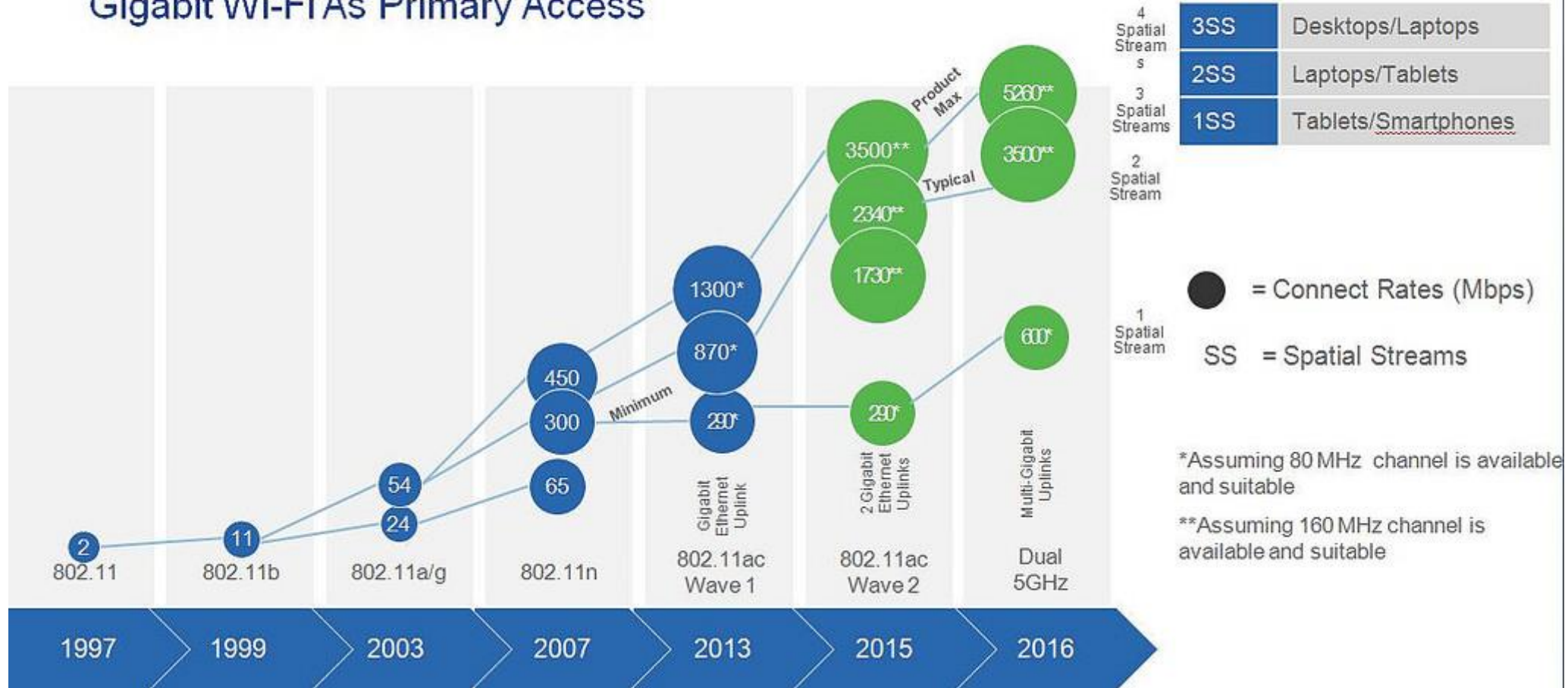
- The standard for wireless networking within a home or office. Also known as a "Wi-Fi" or "802.11" network, wireless Ethernet is the wireless counterpart to regular, wired Ethernet, which is also the standard for local networks. See 802.11 and wireless LAN.

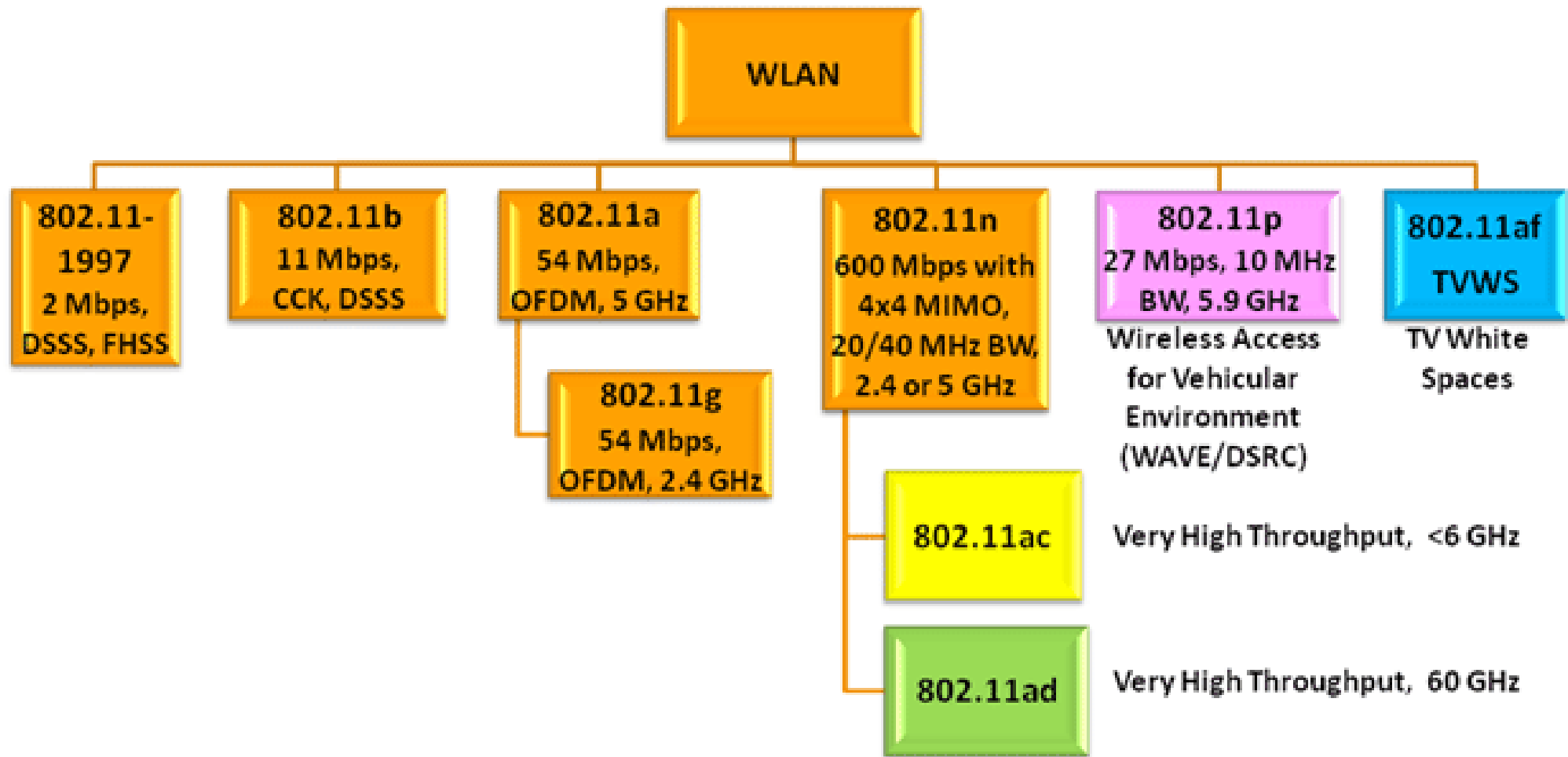


	802.11 (Legacy)	802.11b (Legacy)	802.11a (Legacy)	802.11g (Legacy)	802.11n (HT)	802.11ac (VHT)	802.11ax (HE)
Year Ratified	1997	1999	1999	2003	2009	2014	2019 (Expected)
Operating Band	2.4 GHz/IR	2.4 GHz	5 GHz	2.4 GHz	2.4/5 GHz	5 GHz	2.4/5 GHz
Channel BW	20 MHz	20 MHz	20 MHz	20 MHz	20/40 MHz	20/40/80/160 MHz	20/40/80/160 MHz
Peak PHY Rate	2 Mbps	11 Mbps	54 Mbps	54 Mbps	600 Mbps	6.8 Gbps	10 Gbps
Link Spectral Efficiency	0.1 bps/Hz	0.55 bps/Hz	2.7 bps/Hz	2.7 bps/Hz	15 bps/Hz	42.5 bps/Hz	62.5 bps/Hz
Max # SU Streams	1	1	1	1	4	8	8
Max # MU Streams	NA	NA	NA	NA	NA	4 (DL only)	8 (UL & DL)
Modulation	DSSS, FHSS	DSSS, CCK	OFDM	OFDM	OFDM	OFDM	OFDM, OFDMA
Max Constellation / Code Rate	DQPSK	CCK	64-QAM, 3/4	64-QAM, 3/4	64-QAM, 5/6	256-QAM, 5/6	1024-QAM, 5/6
Max # OFDM tones	NA	NA	64	64	128	512	2048
Subcarrier Spacing	NA	NA	312.5 kHz	312.5 kHz	312.5 kHz	312.5 kHz	78.125 kHz

Wi-Fi Connectivity Speed Timeline

Gigabit Wi-Fi As Primary Access





DSRC = Dedicated Short-Range Communications



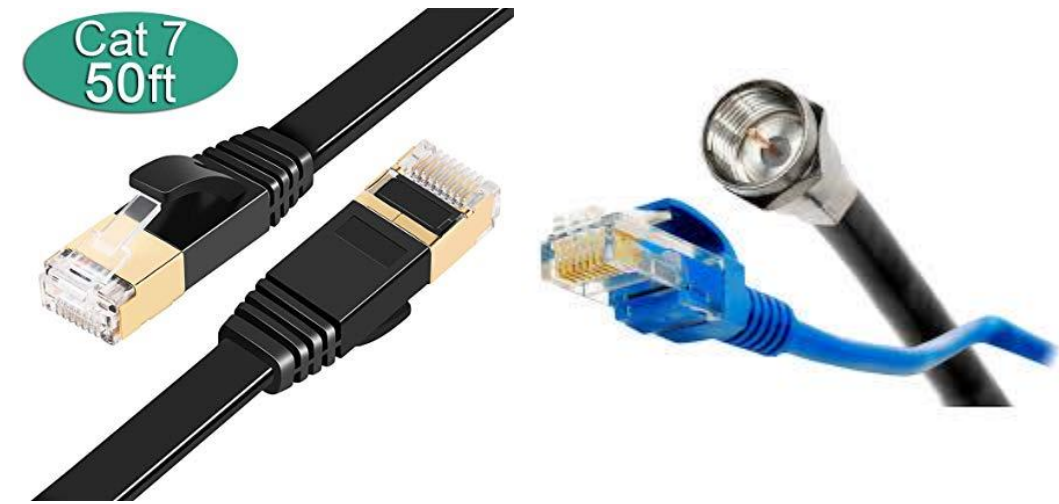
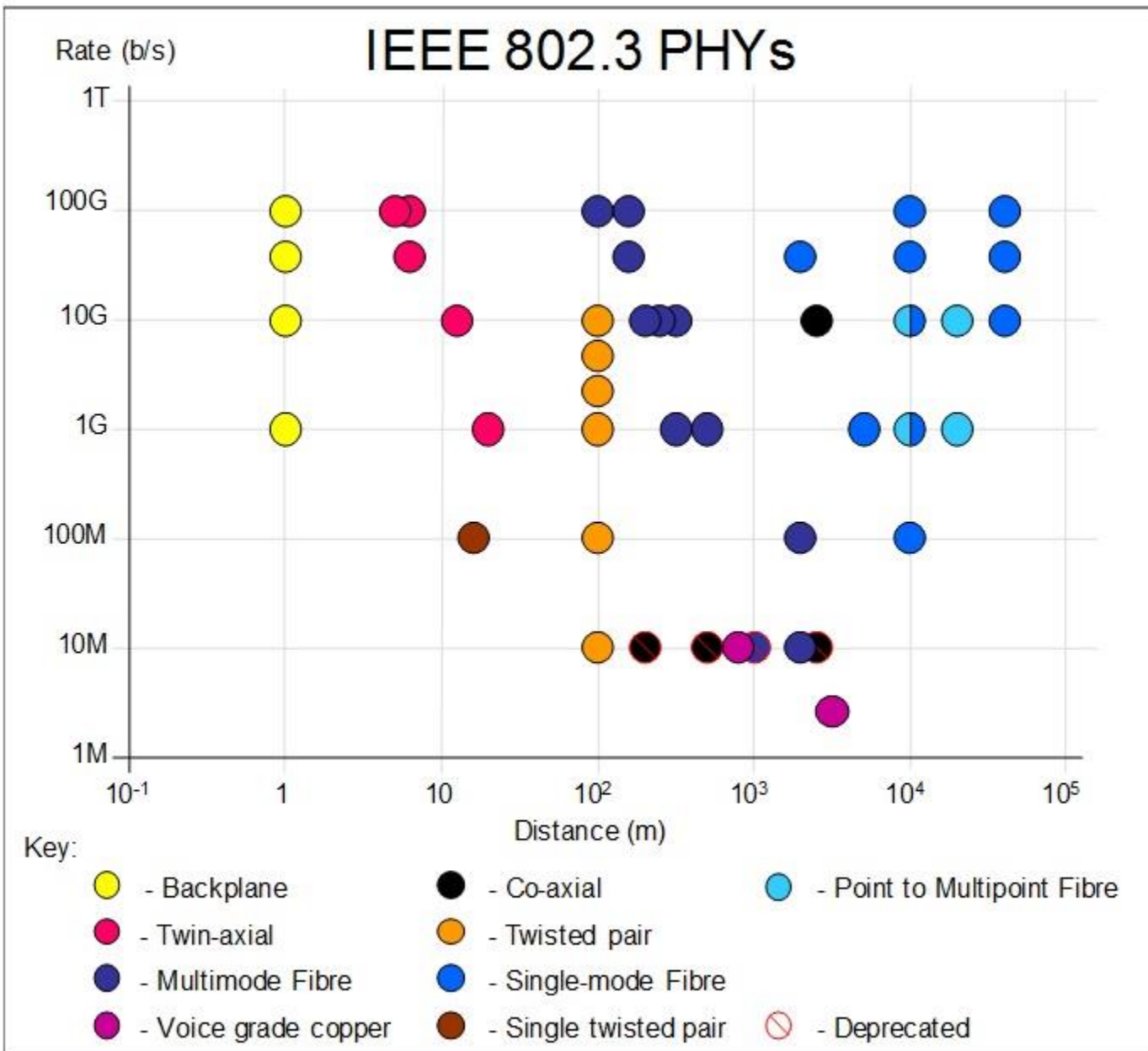
Wired Ethernet

SECTION 7-4

Ethernet Cables and How They Work

- An Ethernet cable is one of the most common forms of network cable used on wired networks. Ethernet cables connect devices within a local area network, like PCs, routers, and switches.
- These physical cables are limited by the distance that they can stretch and still carry proper signals and by their durability.
- These limits are one reason there are different types of Ethernet cables optimized to perform certain tasks in particular situations.

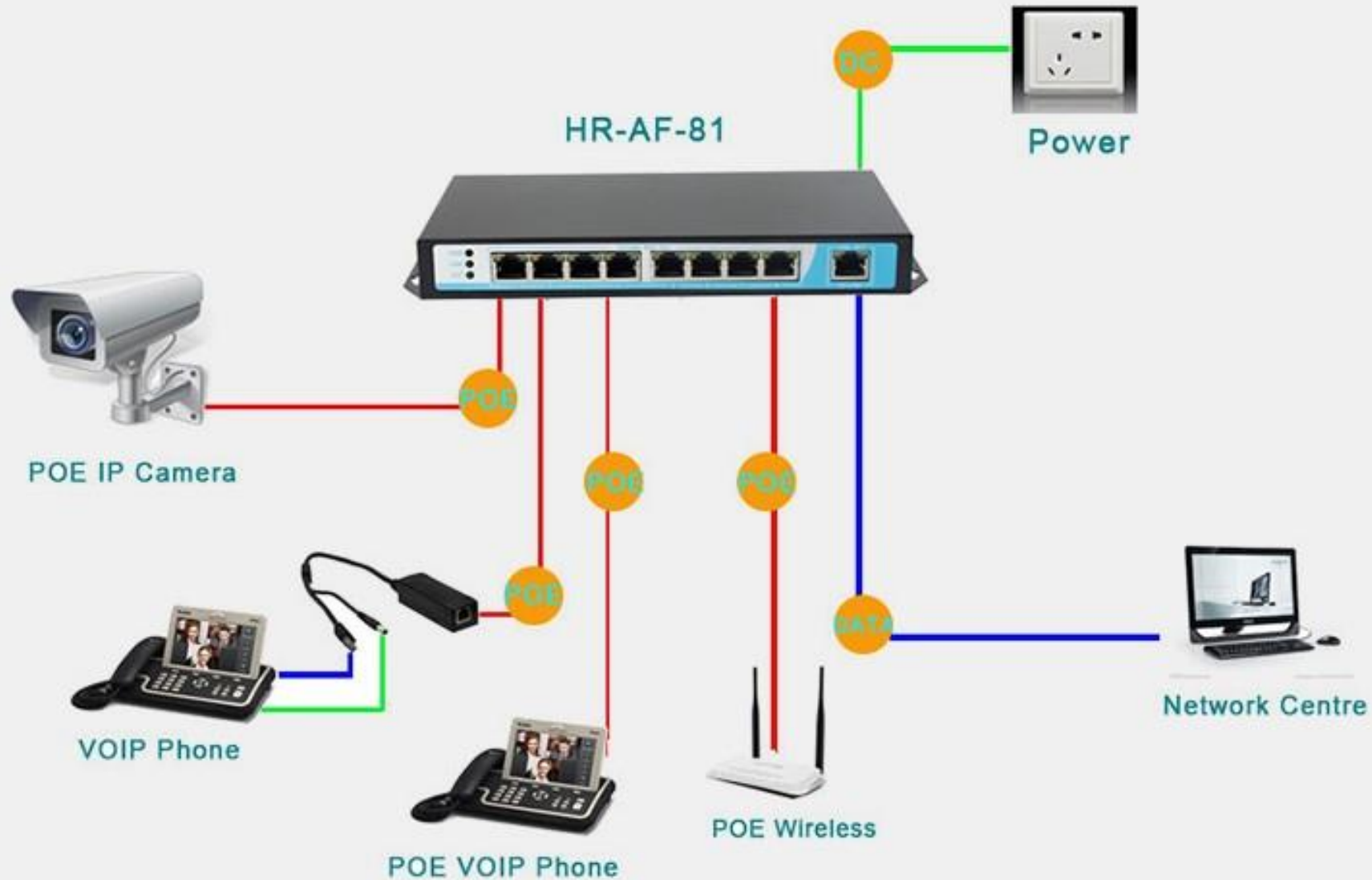
Original IEEE	IEEE Shorthand Name	Informal Name(s)	Speed	Typical Cabling
802.3i	10BASE-T	Ethernet	10 Mbps	UTP
802.3u	100BASE-T	Fast Ethernet (Fast E)	100 Mbps	UTP
802.3z	1000BASE-X	Gigabit Ethernet (Gig E, GbE)	1000 Mbps	Fiber
802.3ab	1000BASE-T	Gigabit Ethernet (Gig E, GbE)	1000 Mbps	UTP
802.3ae	10GBASE-X	10 GbE	10 Gbps	Fiber
802.3an	10GBASE-T	10 GbE	10 Gbps	UTP
802.3ba	40GBASE-X	40GbE (40 GigE)	40 Gbps	Fiber
802.3ba	100GBASE-X	100GbE (100 GigE)	100 Gbps	Fiber



FCoE



Connection Diagram







Power Line Ethernet

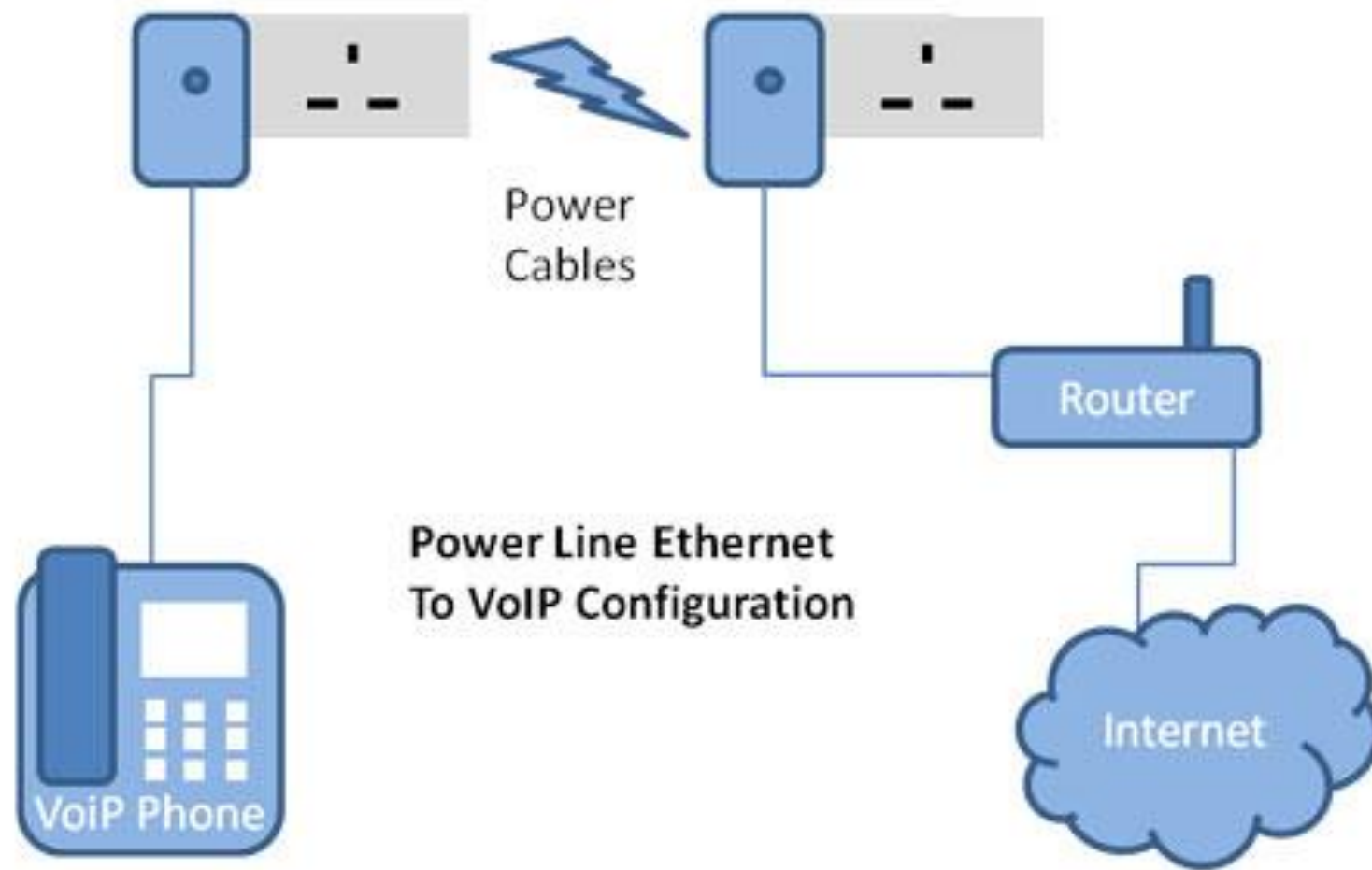
SECTION 7-5

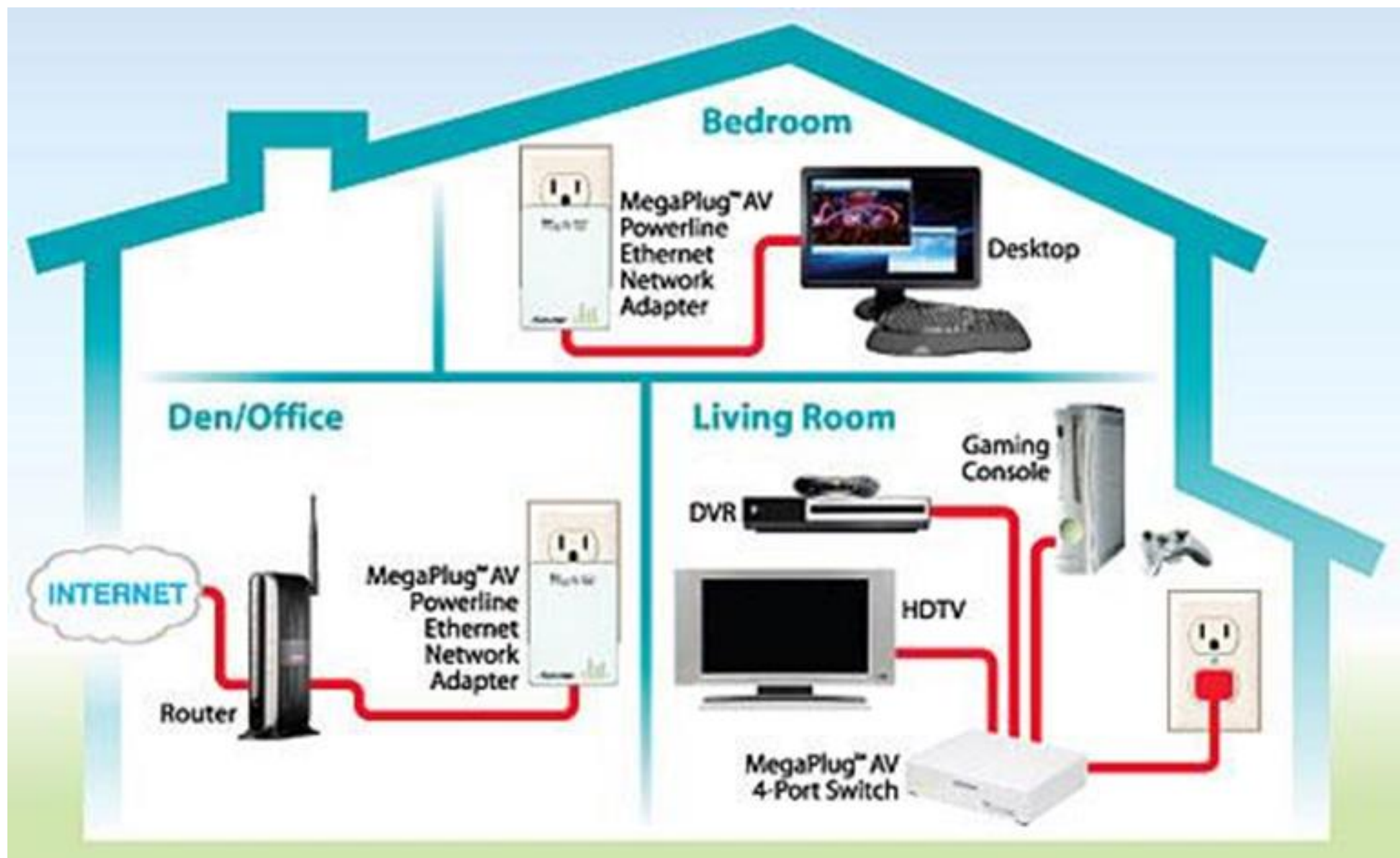
Power-line Communication

- Power-line communication (PLC) carries data on a conductor that is also used simultaneously for AC electric power transmission or electric power distribution to consumers.
- It is also known as power-line carrier, power-line digital subscriber line (PDSL), mains communication, power-line telecommunications, or power-line networking (PLN).

Power-line Communication

- A wide range of power-line communication technologies are needed for different applications, ranging from home automation to Internet access which is often called broadband over power lines (BPL). Most PLC technologies limit themselves to one type of wires (such as premises wiring within a single building), but some can cross between two levels (for example, both the distribution network and premises wiring).
- Typically transformers prevent propagating the signal, which requires multiple technologies to form very large networks. Various data rates and frequencies are used in different situations.





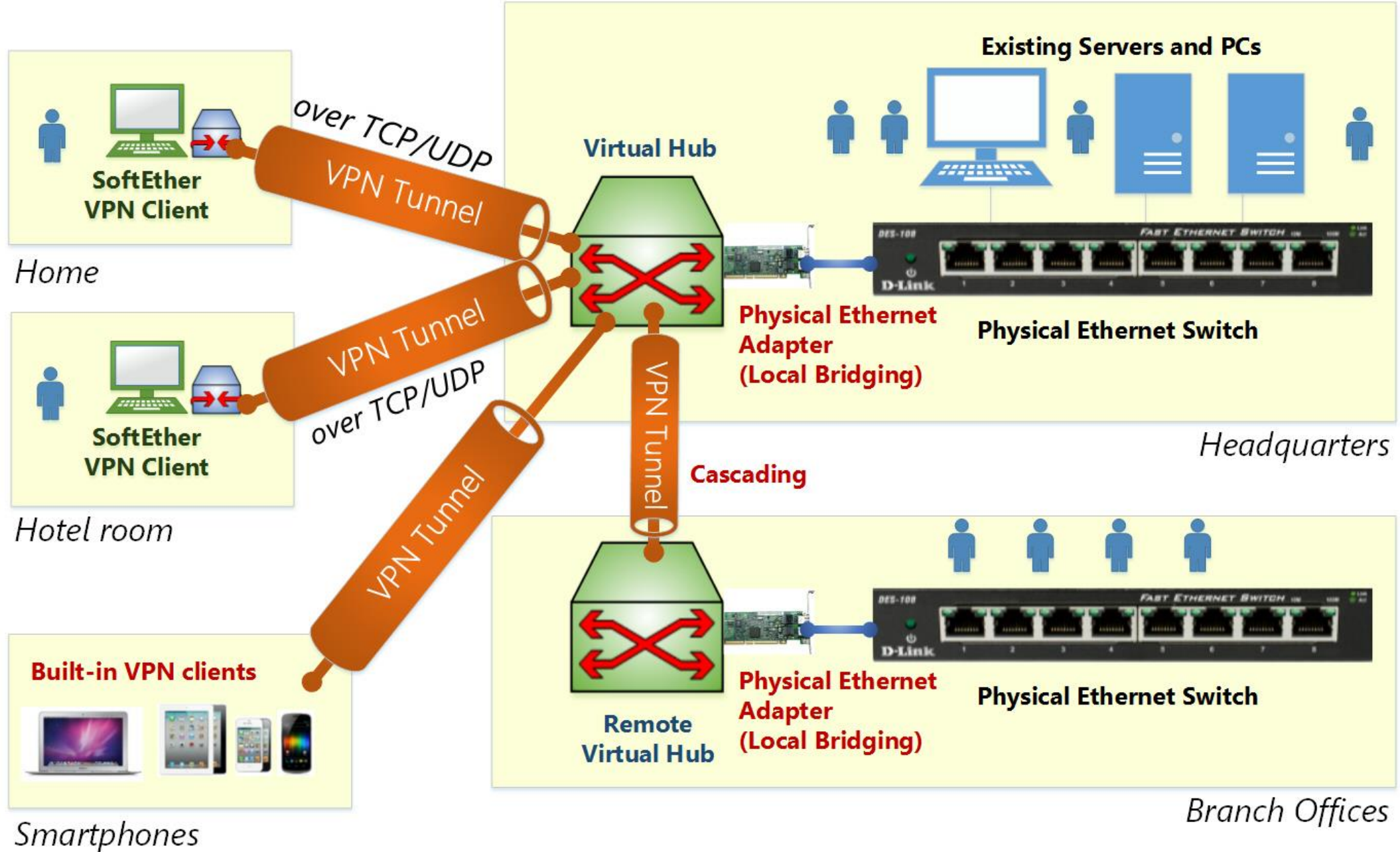


VPN

SECTION 7-6

What Is a VPN? - Virtual Private Network

- A **virtual private network**, or VPN, is an encrypted connection over the Internet from a device to a network.
- The **encrypted** connection helps ensure that sensitive data is safely transmitted. It prevents unauthorized people from eavesdropping on the traffic and allows the user to conduct work remotely.
- VPN technology is widely used in corporate environments.



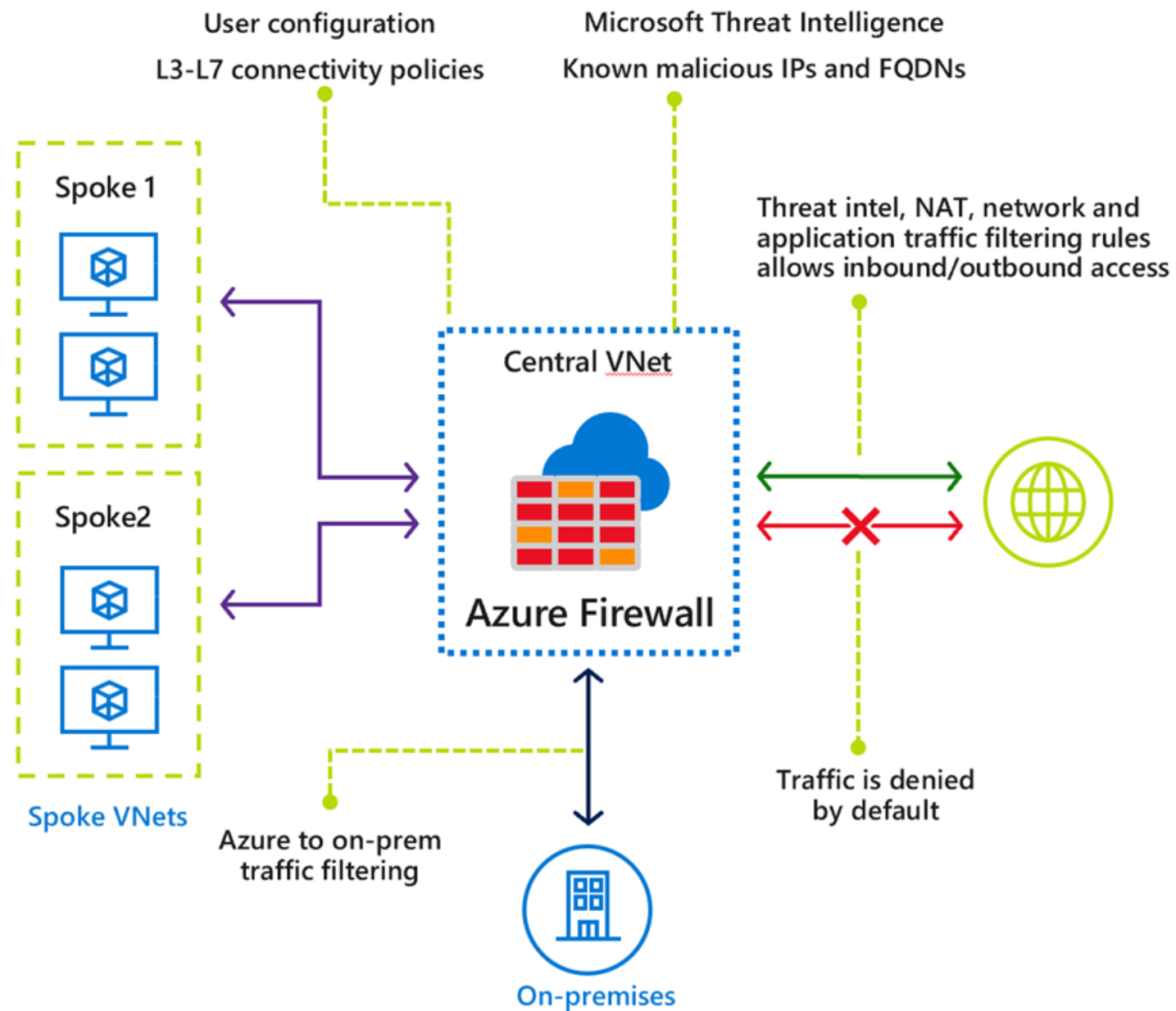


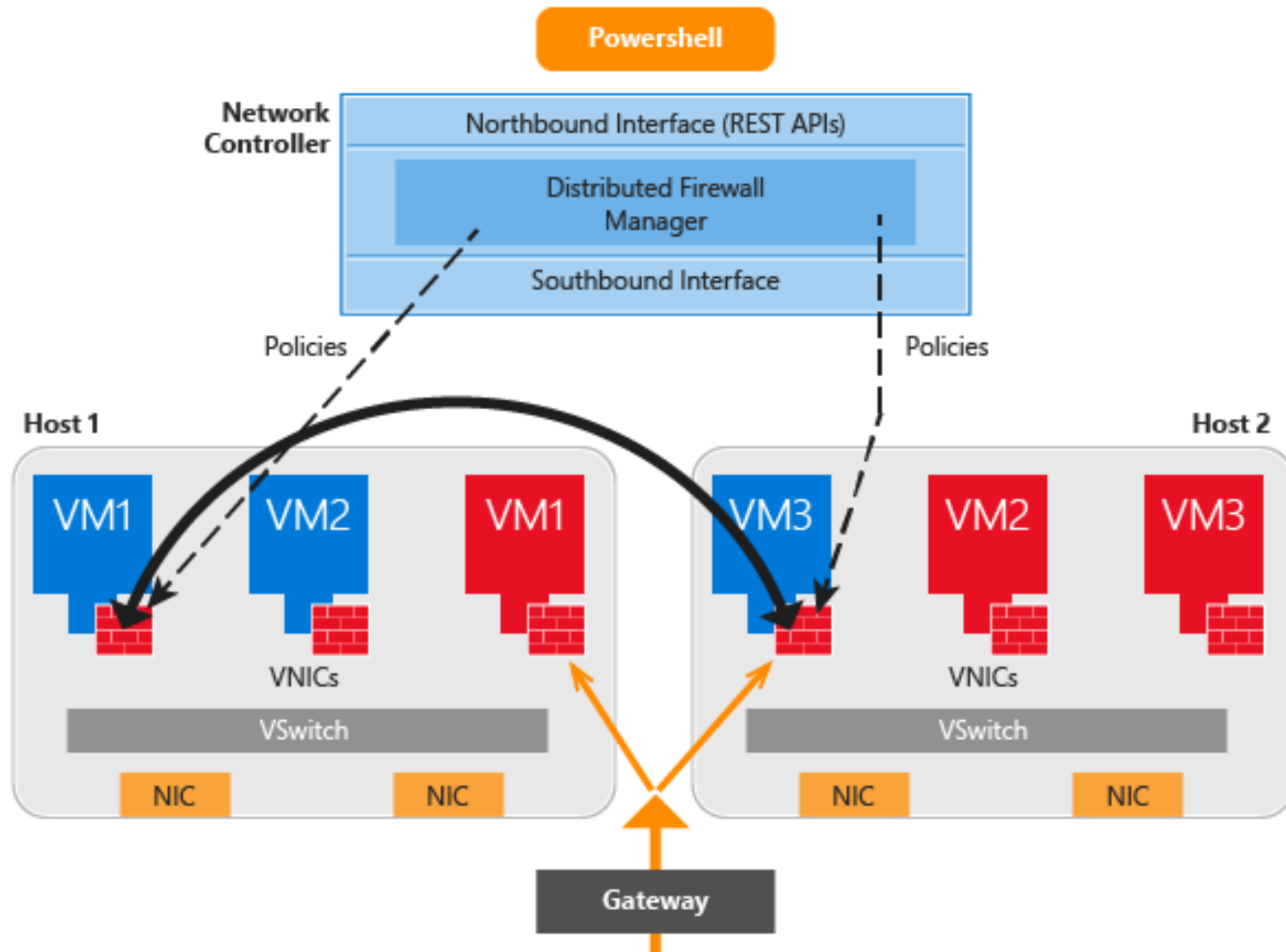
Anti-Virus and Computer Security

LECTURE 8

Firewall

- In computing, a **firewall** is a network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules. A firewall typically establishes a barrier between a trusted internal network and untrusted external network, such as the Internet.
- Firewalls are often categorized as either **network firewalls** or **host-based firewalls**. Network firewalls filter traffic between two or more networks and run on network hardware. Host-based firewalls run on host computers and control network traffic in and out of those machines.





Malware

- **Malware**, or “malicious software,” is an umbrella term that describes any malicious program or code that is harmful to systems.
- Hostile, intrusive, and intentionally nasty, malware seeks to invade, damage, or disable computers, computer systems, networks, tablets, and mobile devices, often by taking partial control over a device’s operations. Like the human flu, it interferes with normal functioning.
- **Malware** is all about making money off you illicitly. Although malware cannot damage the physical hardware of systems or network equipment (with one known exception—see the Google Android section below), it can steal, encrypt, or delete your data, alter or hijack core computer functions, and spy on your computer activity without your knowledge or permission.



