BASIC NETWORK
MQTT - PUB/SUB
Project status
Do prototyping
GET A PROTOTYPE READY

# General Communication

**Synchronous communication:**

Both the sender and the receiver are active at the same time (think of talking on a telephone)

**Asynchronous communication:**

The sending and receiving occur at different times (think of email and answering machines)

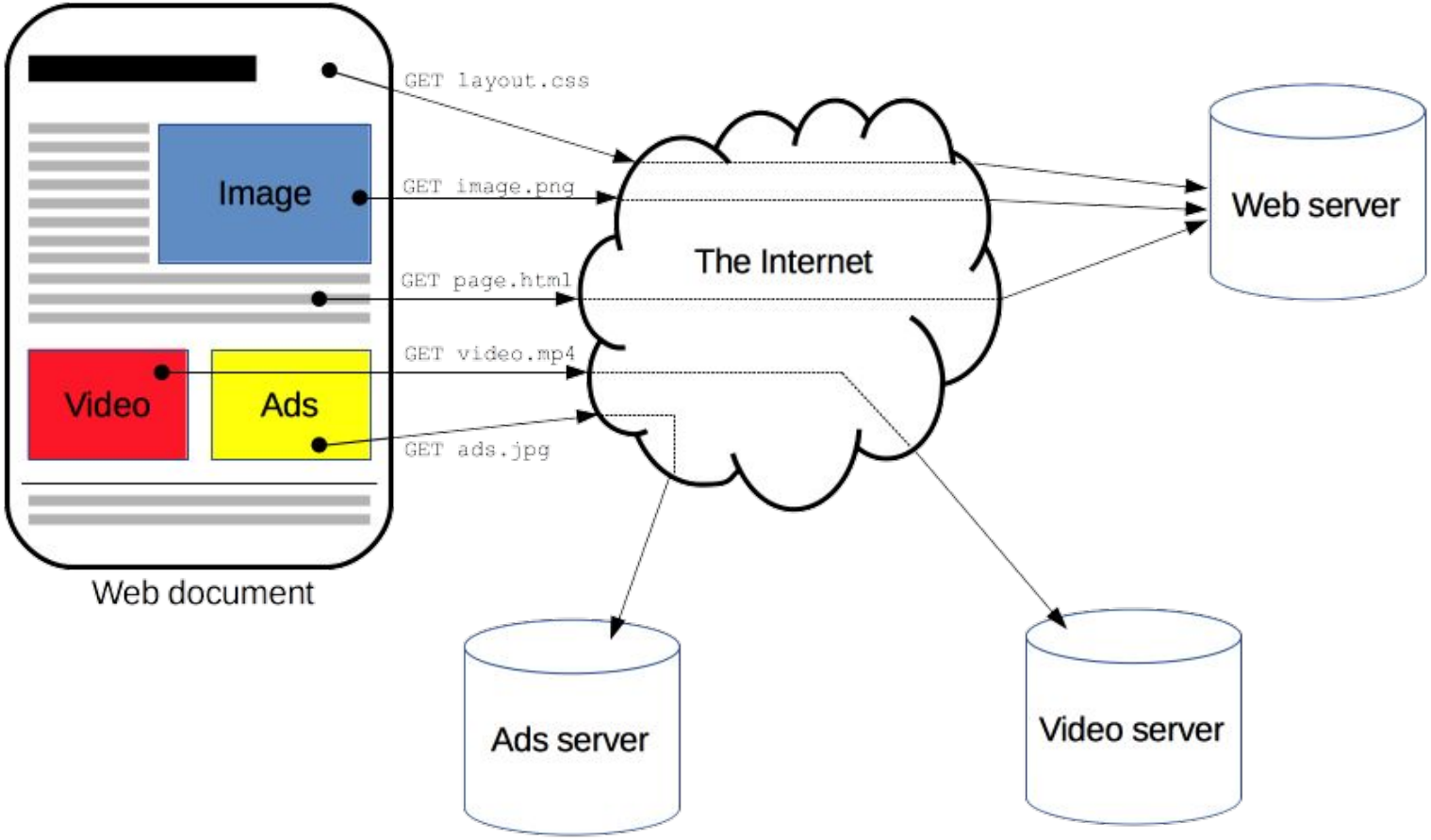
# General Communication

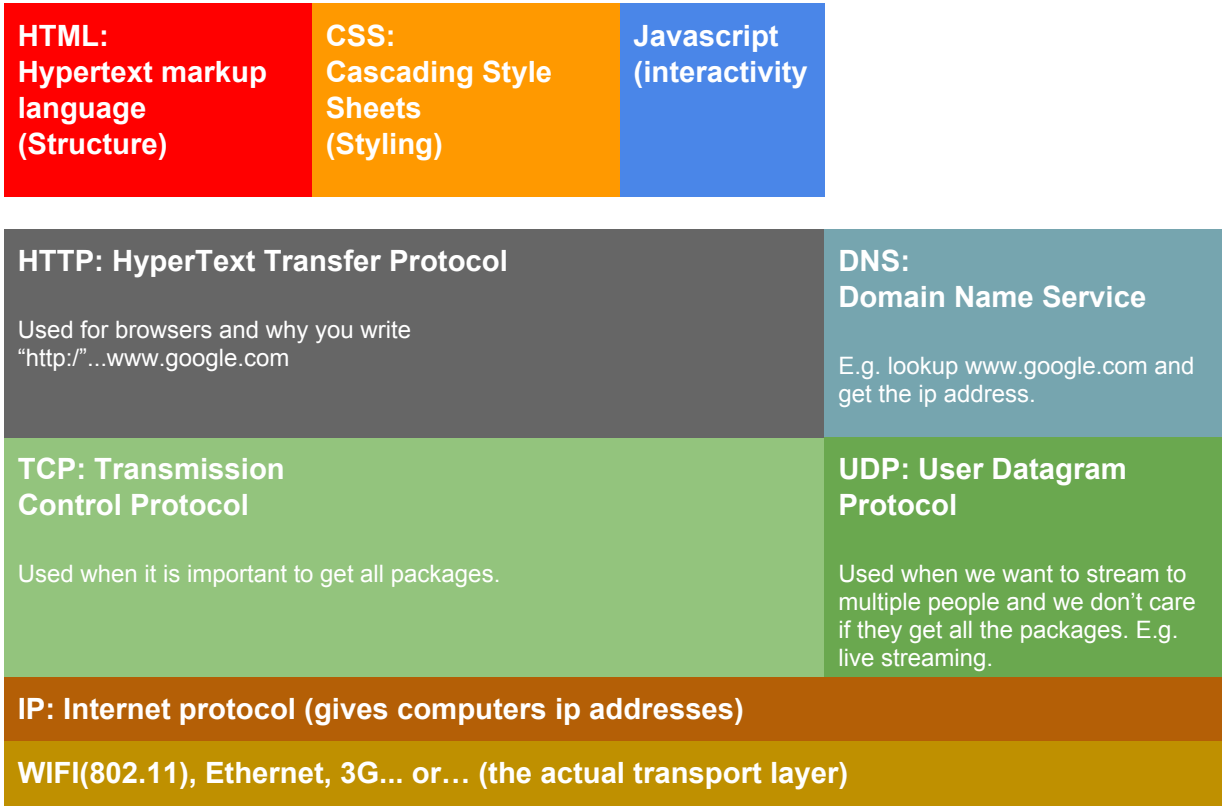
Another property of communication concerns the number of receivers.

**Broadcast communication:** Single sender and many receivers (radio and TV)

**Multicast:** Is many receivers, but usually a specific group (specialized topics)

**Point-to-point communication:** One specific sender and one specific receiver (telephone call)


GET layout.css
Image
GET image.png
GET page.html
GET video.mp4
Video
Ads
GET ads.jpg
Web document
The Internet
Web server
Ads server
Video server




Both TCP and UDP: Separates the messages into pieces and recollect them.

# TRANSPORT LAYER

WIFI(802.11), Ethernet, 3G... or… (the actual transport layer)


1 2 3
4 5 6
7 8 9
* 0 #
Classic
Touch-Tone
Dual
307


NETGEAR

# Ethernet

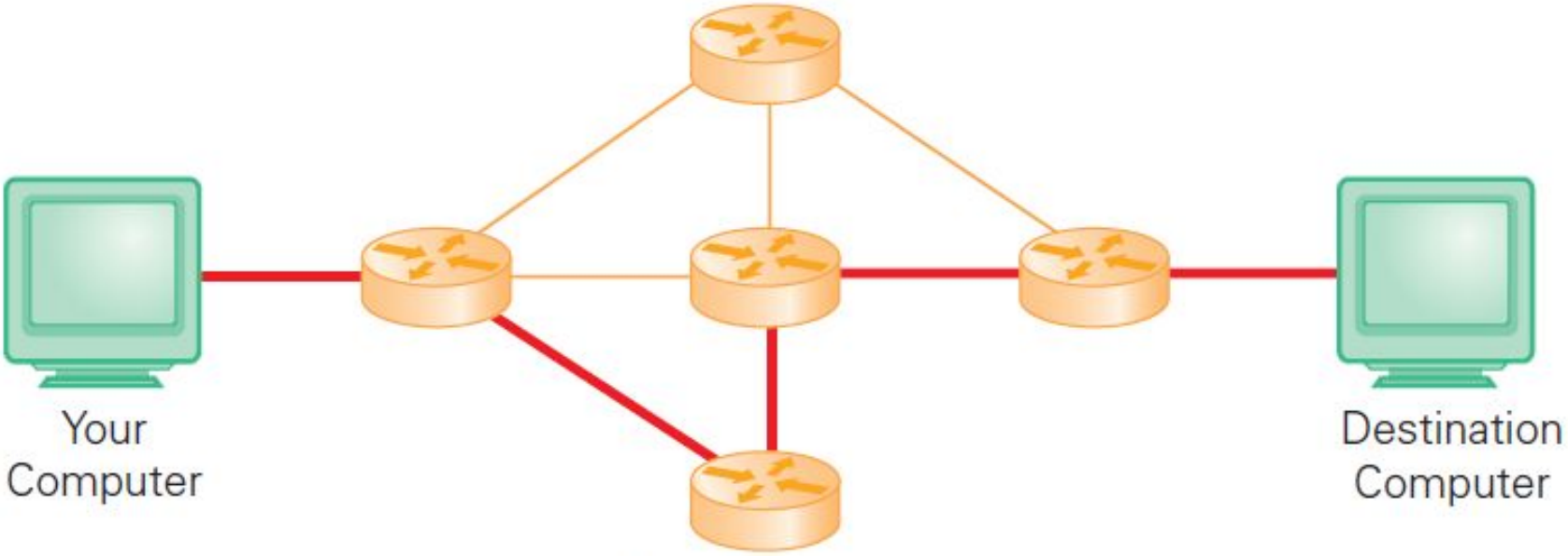
- The physical setup for an Ethernet network is a wire, wire pair, or optical fiber, called the **channel**
- Engineers “tap” into the channel to connect a computer:
  - This allows it to send a signal or an electronic pulse or light flash onto the channel
  - All computers, including the sender, can detect the signal

# Wireless Networks

- Variation of a LAN connection
- Referred to by its protocol name ***802.11***
- The router is:
  - Physically connected to an ISP's modem
  - Connected to the Internet
  - Capable of *broadcasting* **and** *receiving* signals, usually radio frequency (rf ) signals

# Ethernet Party Analogy

- To understand how an Ethernet network works, consider this:
  - A group of friends is standing around at a party telling stories.
  - While someone is telling a story, everyone is listening.
  - When the story is over, here may be a pause before the next one speaks
  - Then, someone typically just begins talking and the cycle starts again


Your
Computer
Destination
Computer
Multiple Routers
(Connections)
Are Involved

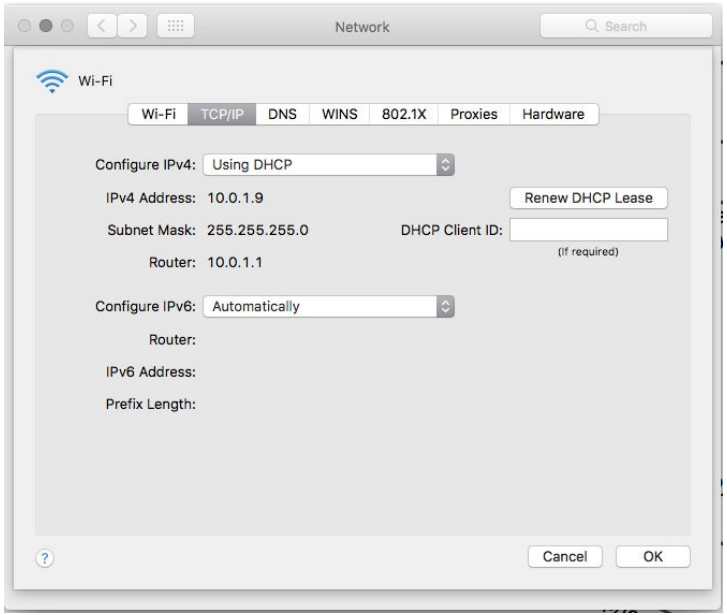
# (IP) INTERNET PROTOCOL

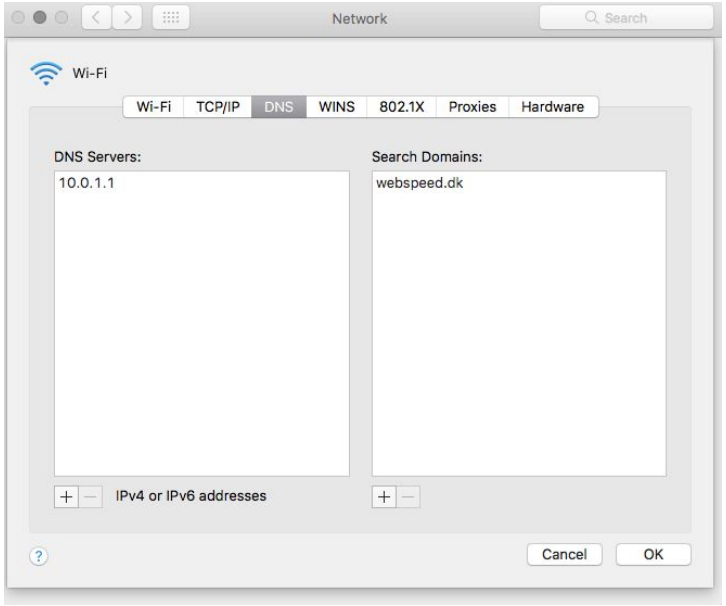
IP: Internet protocol (gives computers ip addresses)

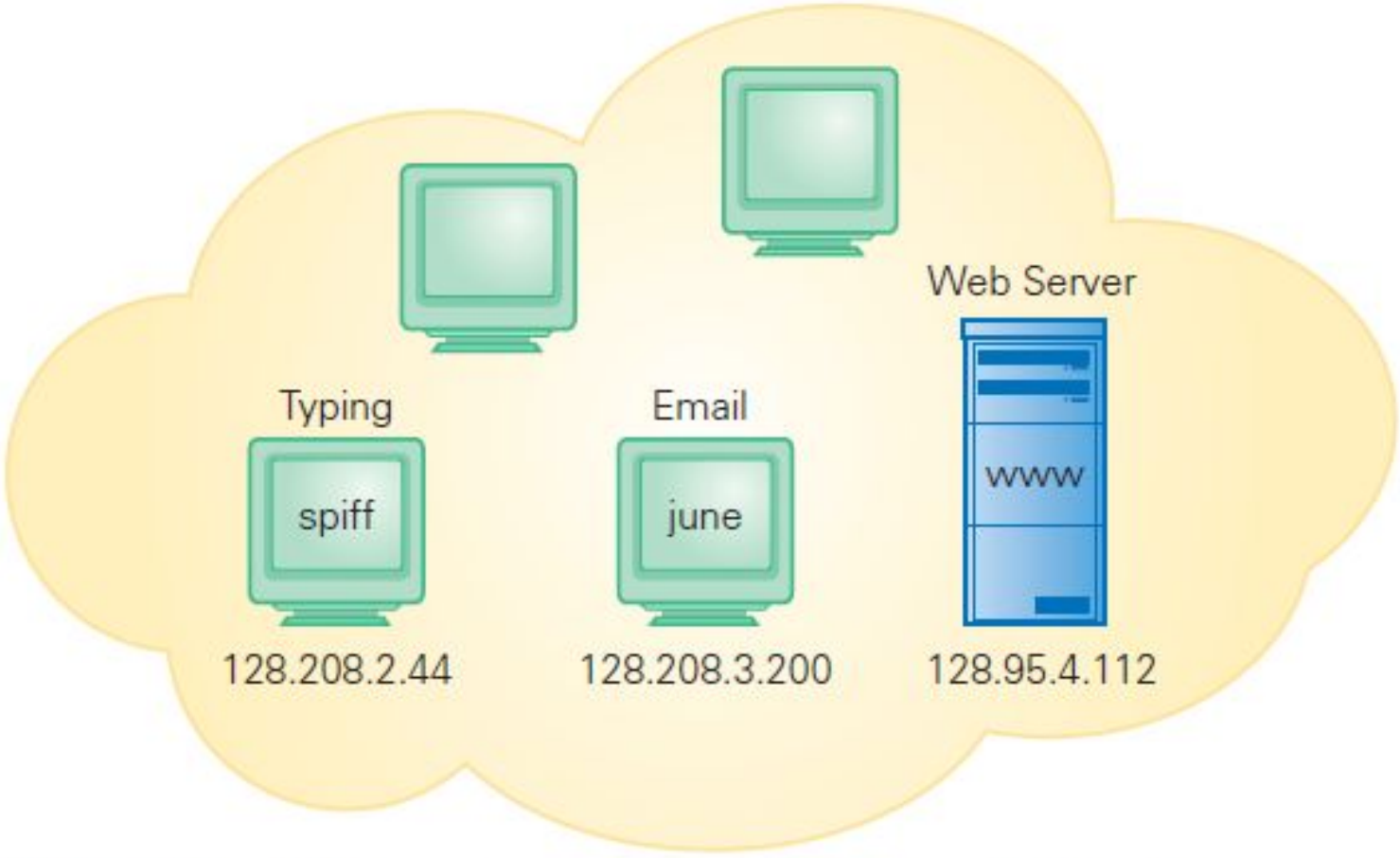
WIFI(802.11), Ethernet, 3G... or… (the actual transport layer)

# Computer Addresses

- IP Addresses
  - Each computer connected to the Internet is given a unique address called its **IP address**
  - An IP address is a series of four numbers (one byte each) separated by dots
  - The range of each of these numbers (0–255) allows for billions of IP addresses
  - New IP addresses are in short supply

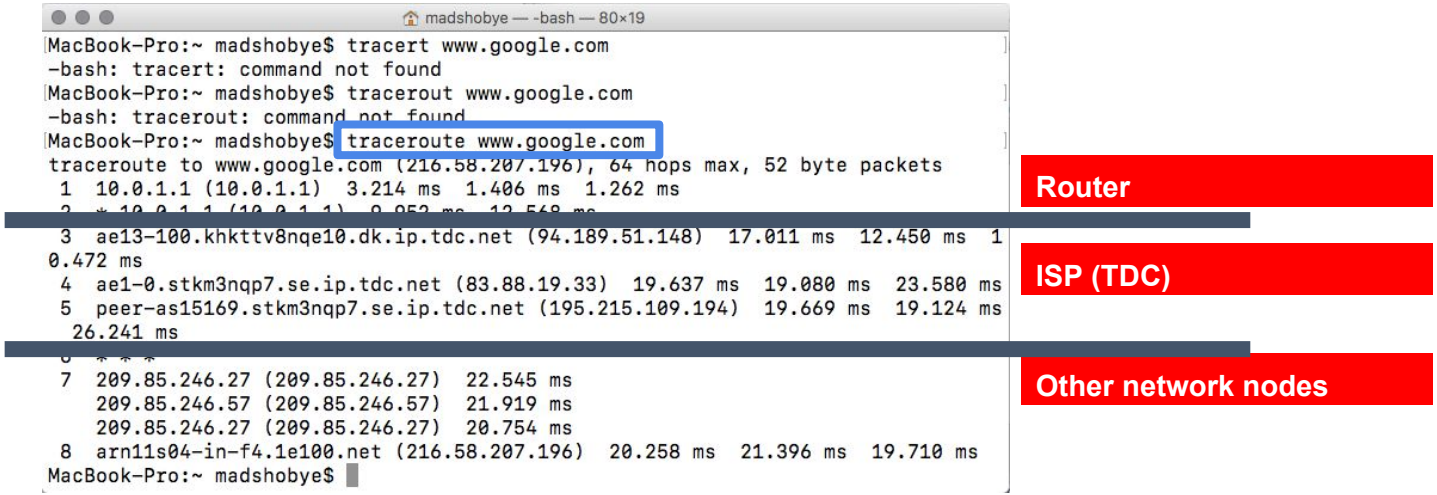

Network
Search
Wi-Fi
Wi-Fi
TCP/IP
DNS
WINS
802.1X
Proxies
Hardware
Configure IPv4:
Using DHCP
IPv4 Address:
10.0.1.9
Renew DHCP Lease
Subnet Mask:
255.255.255.0
DHCP Client ID:
(If required)
Router:
10.0.1.1
Configure IPv6:
Automatically
Router:
IPv6 Address:
Prefix Length:
Cancel
OK


Network
Search
Wi-Fi
Wi-Fi
TCP/IP
DNS
WINS
802.1X
Proxies
Hardware
DNS Servers:
10.0.1.1
Search Domains:
webspeed.dk
IPv4 or IPv6 addresses
Cancel
OK


Web Server
Typing
Email
www
spiff
june
128.208.2.44
128.208.3.200
128.95.4.112

# Moving Packets: Wires & More

- Internet uses telephone carriers for long-distance connections, fiber optics, and separate dedicated lines for connections
- The computers do not know or care how the packet is sent, as long as it can be sent and received
- Transmissions may rely on multiple technologies as the packets move across the Internet


madshobye — -bash — 80×19
MacBook-Pro:~ madshobye$ tracert www.google.com
-bash: tracert: command not found
MacBook-Pro:~ madshobye$ tracerout www.google.com
-bash: tracerout: command not found
MacBook-Pro:~ madshobye$ traceroute www.google.com
traceroute to www.google.com (216.58.207.196), 64 hops max, 52 byte packets
1 10.0.1.1 (10.0.1.1) 3.214 ms 1.406 ms 1.262 ms
3 ae13-100.khkttv8nqe10.dk.ip.tdc.net (94.189.51.148) 17.011 ms 12.450 ms 1
0.472 ms
4 ae1-0.stkm3nqp7.se.ip.tdc.net (83.88.19.33) 19.637 ms 19.080 ms 23.580 ms
5 peer-as15169.stkm3nqp7.se.ip.tdc.net (195.215.109.194) 19.669 ms 19.124 ms
26.241 ms
7 209.85.246.27 (209.85.246.27) 22.545 ms
209.85.246.57 (209.85.246.57) 21.919 ms
209.85.246.27 (209.85.246.27) 20.754 ms
8 arn11s04-in-f4.1e100.net (216.58.207.196) 20.258 ms 21.396 ms 19.710 ms
MacBook-Pro:~ madshobye$
Router
ISP (TDC)
Other network nodes

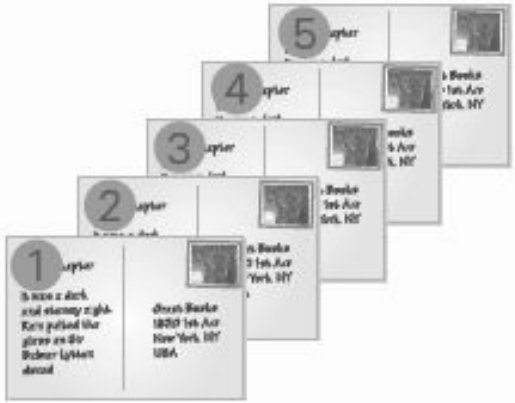
# (TCP) TRANSMISSION CONTROL PROTOCOL

**TCP: Transmission Control Protocol**

Used when it is important to get all packages.

**IP: Internet protocol (gives computers ip addresses)**

**WIFI(802.11), Ethernet, 3G... or… (the actual transport layer)**


5
4
3
2
1


Where's
#17?

# TCP/IP

- TCP/IP Postcard Analogy
  - The Internet is like sending a novel to your publisher using postcards
  - The novel is broken into small units that fit on a postcard
  - The “postcards” are numbered to indicate where each belongs in the novel
  - As each postcard is completed, it is mailed

# TCP/IP

- TCP/IP Postcard Analogy
  - Sooner or later, your publisher received the postcards, but not necessarily in sequential order
  - Nor do they take the same route
  - The cards are finally arranged in order
  - These “postcards” are really ***IP packets***
    - They hold: one unit of information, the destination IP, and their sequence number (which packet they are)

# Packets Are Independent

- Because each packet can take a different route, congestion and service interruptions do not delay transmissions
  - Each TCP/IP packet is *independent*
- The TCP/IP protocol works under adverse conditions
  - If traffic is heavy and the packet progress is slow, the protocol allows the packet to be thrown away
- If a packet is killed for whatever reason, the recipient will request a resend
- Packets can arrive out of order because they take different routes

- https://www.youtube.com/watch?v=xIuBmOufbls
- https://www.youtube.com/watch?v=i5oe63pOhLI
- https://www.youtube.com/watch?v=7_LPdttKXPc&list=PL847B0678C0E6F908

# (UDP) USER DATAGRAM PROTOCOL

**UDP: User Datagram Protocol**

Used when we want to stream to multiple people and we don't care if they get all the packages. E.g. live streaming.

**IP: Internet protocol (gives computers ip addresses)**

**WIFI(802.11), Ethernet, 3G... or… (the actual transport layer)**

UDP is like TCP but without:

- Resending of lost packages.
- No confirmation of arrival.
- Able to do multicast (sending the same package to multiple users)

# (DNS)
# DOMAIN NAME SERVER

**DNS:**
**Domain Name Service**

E.g. lookup www.google.com and get the ip address.

**UDP: User Datagram Protocol**

Used when we want to stream to multiple people and we don't care if they get all the packages. E.g. live streaming.

Both TCP and UDP: Separates the messages into pieces and recollect them.

**IP: Internet protocol (gives computers ip addresses)**

**WIFI(802.11), Ethernet, 3G... or… (the actual transport layer)**

# DNS Servers

- When your computer asks a DNS name server to translate a name to the IP address, it is in another client/server relationship
- If the address is new (and not stored on the DNS server), the server asks an **authoritative name server**
- A root server keeps *the* complete list of the IP addresses and corresponding domain names for all authoritative name servers and computers in its domain
- They are listed at www.rootservers.org together with their mirror sites (helper name servers with identical information).
- Notice that computers change their client and server roles all the time. Sometimes they are servers, sometimes they are clients.

# Computer Addresses

- Domain Names
  - It is hard to remember the numeric IP address of all the computers we communicate with
  - The Internet uses human-readable symbolic names for computers that are based on a hierarchy of *domains*
  - A **domain** is a related group of networked computers

# DNS Servers

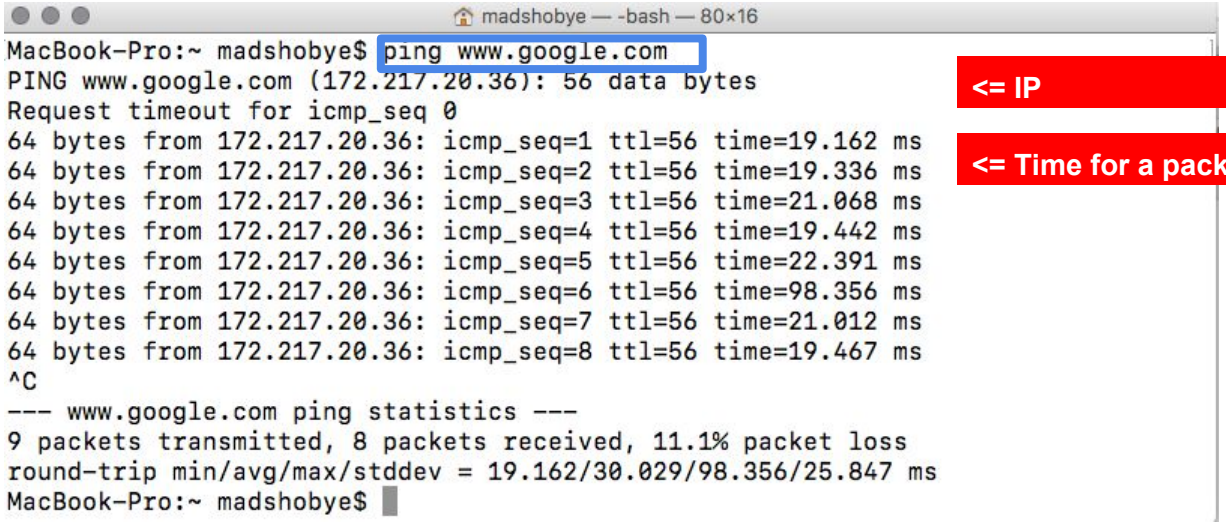
- The **Domain Name System (DNS)** translates the hierarchical, human-readable names into the four-number IP address
- Every Internet host knows the IP address of its nearest **DNS name server**
- Whenever the hierarchical symbolic name is used to send information to a destination, your computer asks the DNS server which looks up the corresponding IP address

1.user types url:
http://www.google.com
in browser

2. Ask DNS server (via udp/tcp) for ip for www.google.com

3. DNS server returns (via udp/tcp) ip 216.58.207.196

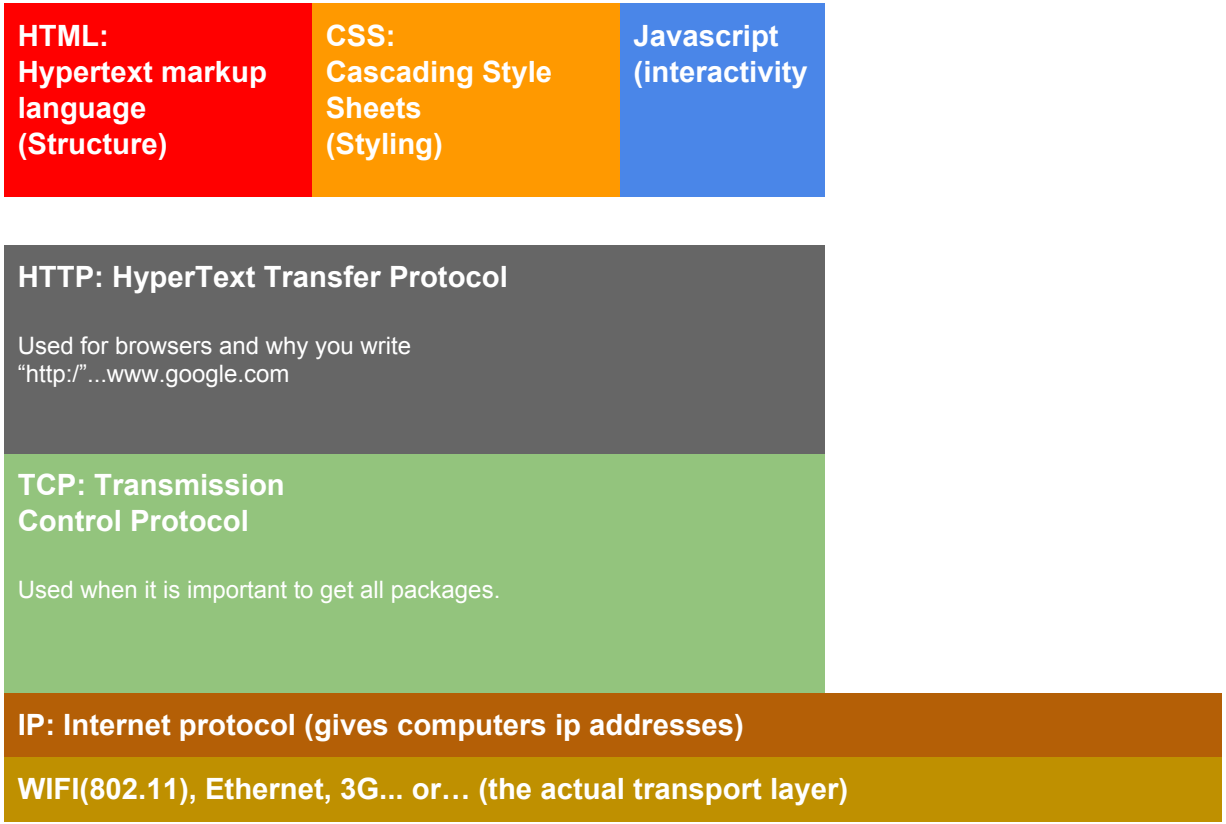
DNS Server




<= IP

<= Time for a package to return

# (HTTP) HYPERTEXT TRANSFER PROTOCOL


HTML:
Hypertext markup language
(Structure)
CSS:
Cascading Style Sheets
(Styling)
Javascript
(interactivity
HTTP: HyperText Transfer Protocol
Used for browsers and why you write
“http:/”...www.google.com
TCP: Transmission
Control Protocol
Used when it is important to get all packages.
IP: Internet protocol (gives computers ip addresses)
WIFI(802.11), Ethernet, 3G... or… (the actual transport layer)


N
NE
E
SE
S
SW
W
NW

# Hypertext Transfer Protocol

- A high level protocol built on top of a TCP connection for exchanging messages (with *arbitrary content)*
  - Each (request) message from client to server is followed by a (response) message from server to client.
  - Facilitates the remote invocation of methods on the server.
- Web: A set of client and server processes on the Internet that communicate via HTTP.

# Client/Server Structure

- Most interactions over the Internet use the *client/server interaction* protocol:
  - When you click a Web link, your computer gets the page for you...beginning the client/server interaction
    - Your computer is the *client* computer and the computer with the Web page is the *server* (**Web server**)
    - The *client,* gets services from the *server*
  - When the page is returned, the operation is completed and the client/server relationship ends

# Basic Client/Server Interaction

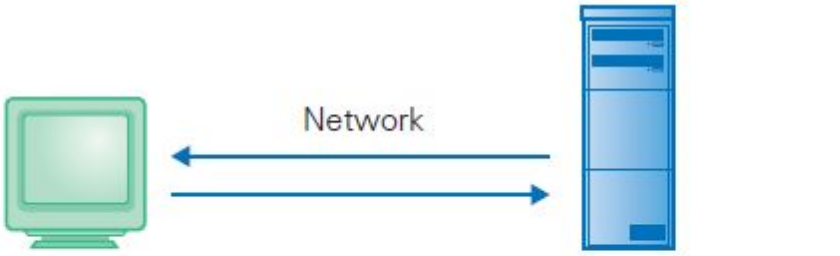



Figure 3.2 The basic client/server interaction, as illustrated by the browser (client) requesting Web pages provided by the Web server.

# Many Brief Relationships

- The client/server structure is fundamental to Internet interactions
- A key aspect for many connection is that only a single service request and response are involved
- This approach means that the server can handle many clients at a time
- For example, between two consecutive client requests from your browser (getting a page and asking for another) that server could have serviced hundreds of other clients
- The server is busy only for as long as it takes to perform your request

# Requesting a Web Page

- Web requests use client/server interaction
- Requesting a Web page means your browser is a *client* asking for a file from a Web *server*
- The file can be found in looking at the ***URL*** (Universal Resource Locator)
- Web browsers and Web servers both “speak” *HTTP*

# Requesting a Web Page

**http**://**www.cs.washington.edu**/**homes/snyder/index.html**

- The **URL** has three main parts:
  - **Protocol.**
    **tells the computers how to handle the file**
  - **Server computer's name**
    or the name given by the domain hierarchy
  - **Page's pathname.**
    tells the server which file (page) is requested and where to find it

1.user types url:
http://www.google.com
in browser

6. Browser makes a http request to ip 216.58.207.196 (via tcp/ip) for images, css etc..

5. Server return HTML page (via tcp/ip) to browser

4. Browser makes a http request (via tcp/ip) to ip 216.58.207.196

ip: 216.58.207.196 (google server)

# Client/Server Interaction

1. User supplies URL (clicks on link)

   http://yourbusiness.com/~items/printers.html

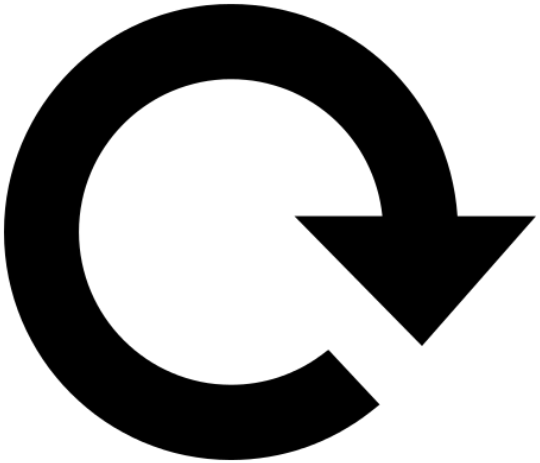
2. Browser translates <*host_name*> *(yourbusiness.com)* to *host_ip_address (using name server)*

4. Browser sets up TCP connection to yourbusiness.com at *(host internet address)*

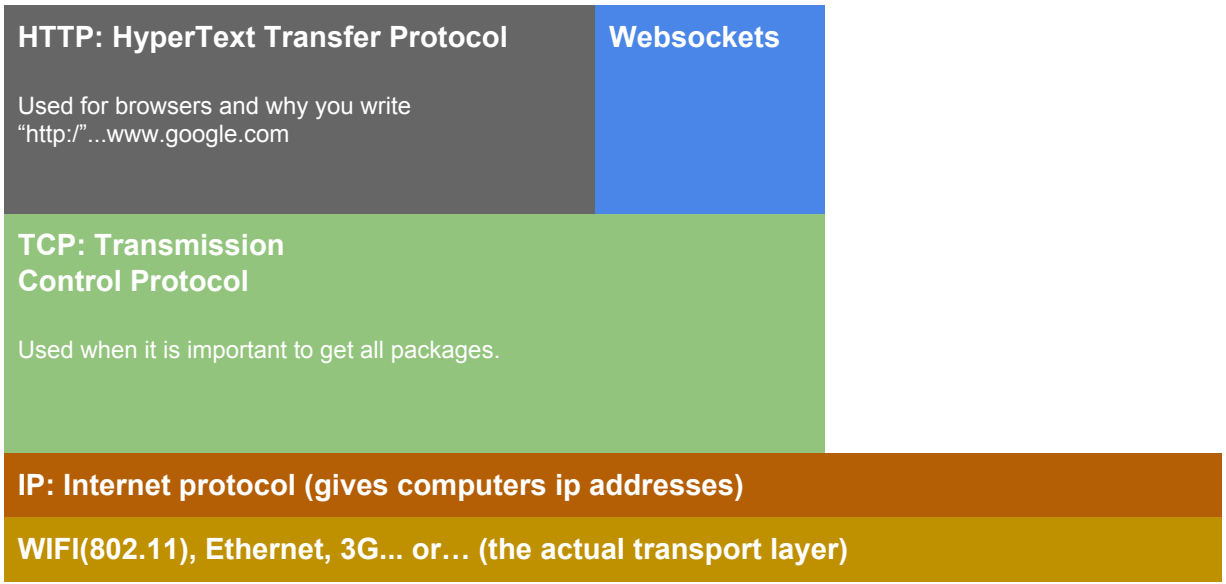
5. Browser sends http message

   GET ~items/printers.html HTTP/1.0 over connection

# CLASSIC HTTP: PULL



Requires refresh for updates
(websocket has solved this problem)


HTTP: HyperText Transfer Protocol
Used for browsers and why you write “http:/”...www.google.com
Websockets
TCP: Transmission Control Protocol
Used when it is important to get all packages.
IP: Internet protocol (gives computers ip addresses)
WIFI(802.11), Ethernet, 3G... or… (the actual transport layer)

# (MQTT)
# MESSAGE QUEUING
# TELEMETRY
# TRANSPORT

MQTT = Message Queuing Telemetry Transport

- Publish-subscribe-based messaging protocol.
- It works on top of the TCP/IP protocol (can also work on other protocols).
- Small code footprint.
- For limited Network bandwidth
- Requires a message broker.

**MQTT: Message Queuing Telemetry Transport**

**TCP: Transmission Control Protocol**

Used when it is important to get all packages.

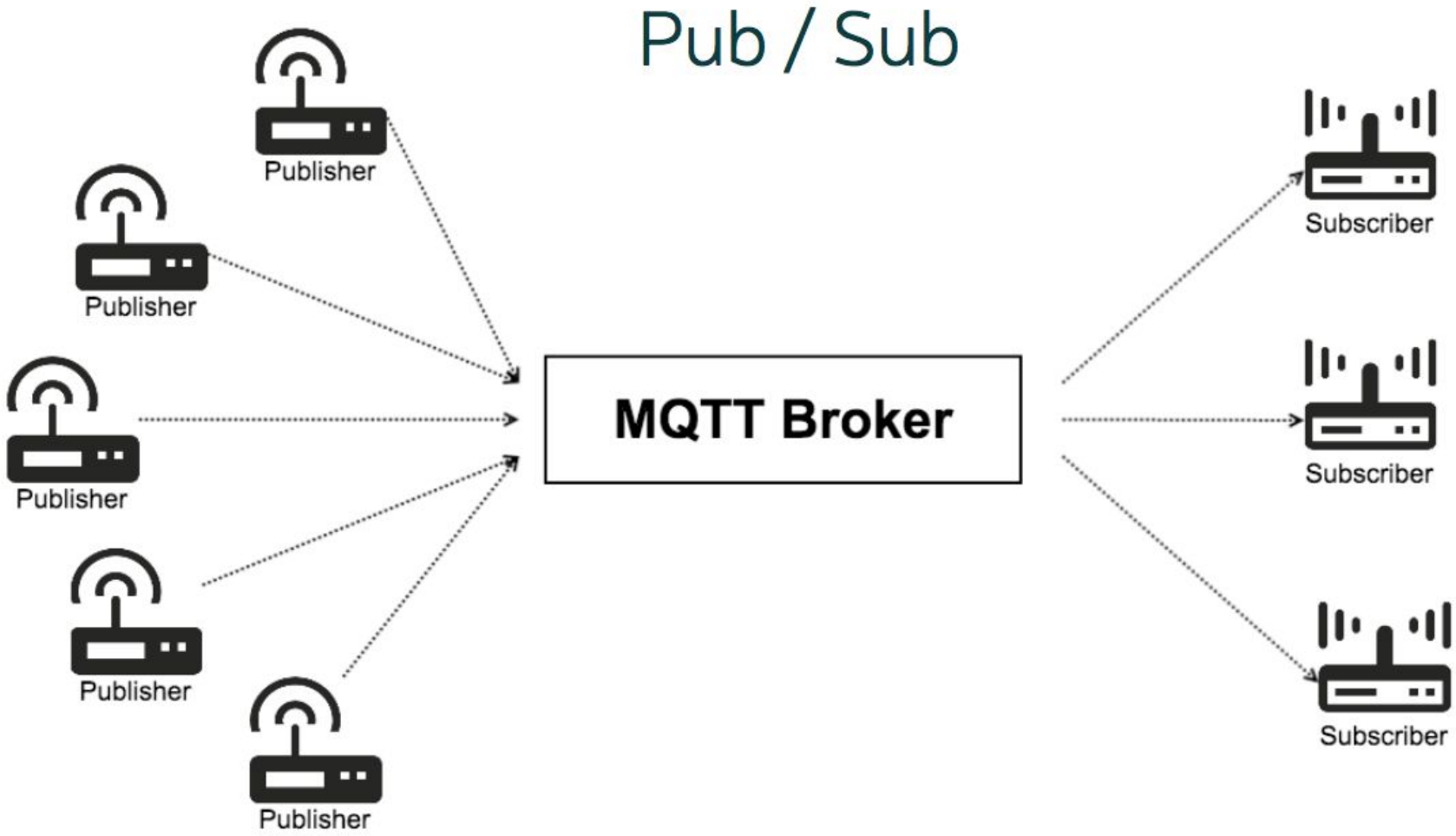
**IP: Internet protocol (gives computers ip addresses)**

**WIFI(802.11), Ethernet, 3G... or… (the actual transport layer)**

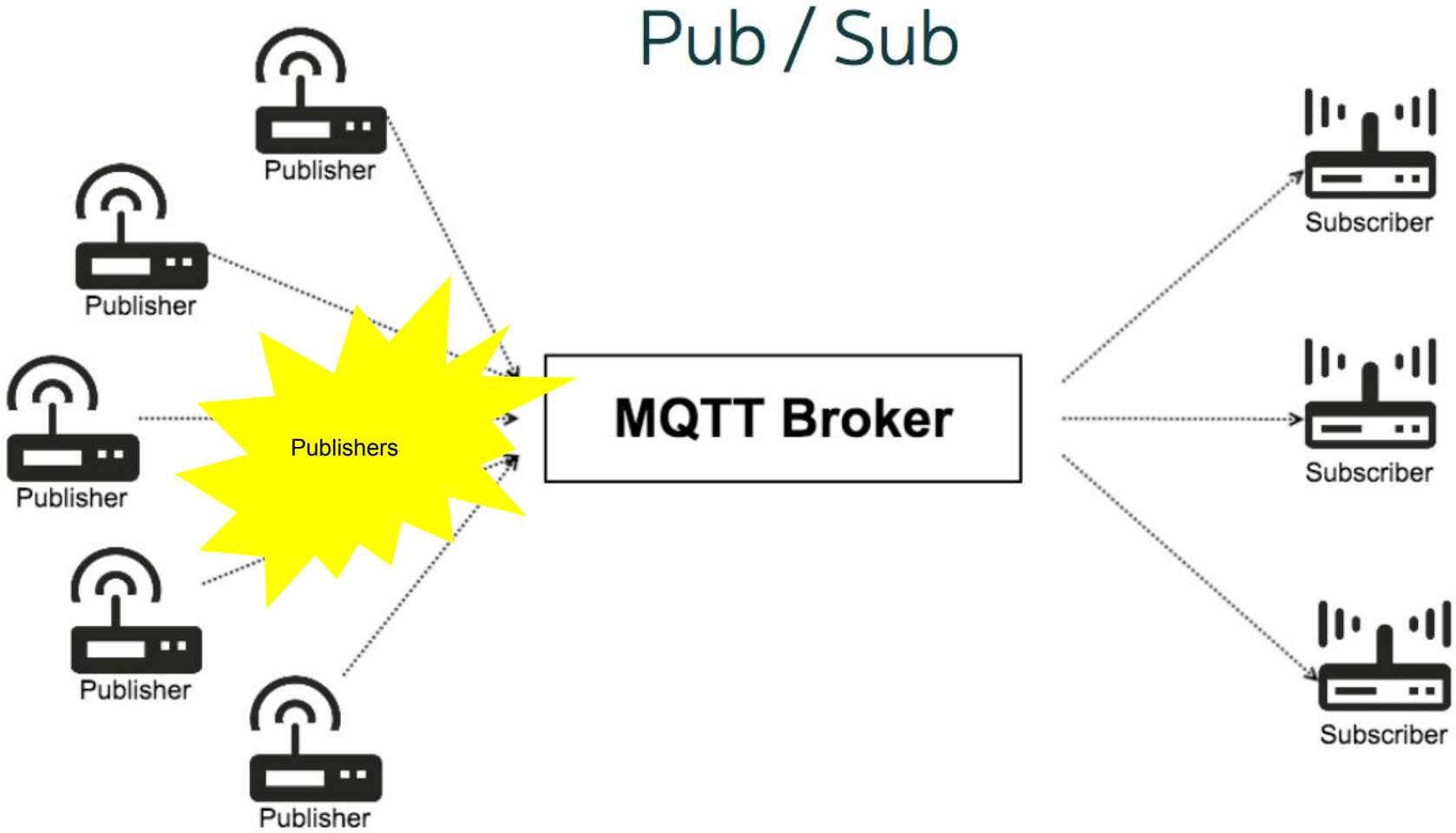
Publish-Subscribe messaging pattern

- Senders of messages are called publishers.
- Receivers are called subscribers.
- Senders do not sent directly to specific receivers.
- Senders categorize published messages into classes.
- Subscribers express interest (subscribes to) in classes.

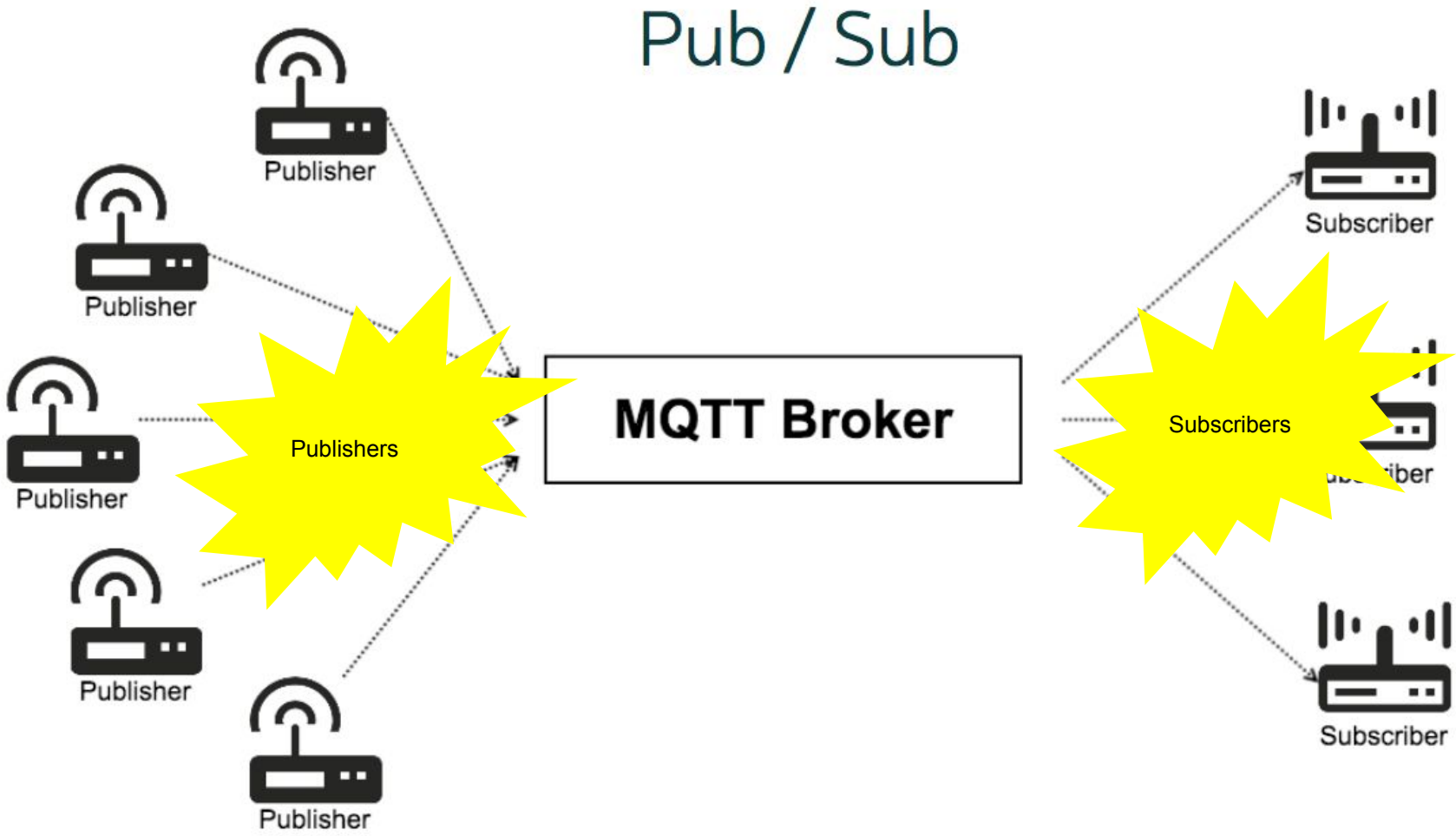
PUSH BASED

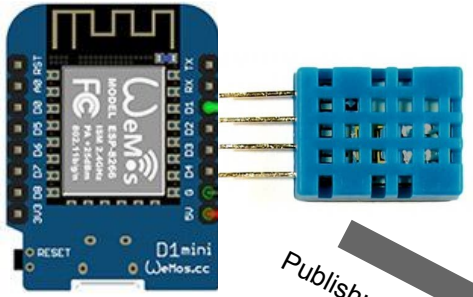

Pub / Sub
Publisher
Publisher
Publisher
Publisher
Publisher
MQTT Broker
Subscriber
Subscriber
Subscriber

PUSH BASED


Pub / Sub
Publisher
Publisher
Publisher
Publisher
Publisher
Publishers
MQTT Broker
Subscriber
Subscriber
Subscriber

PUSH BASED

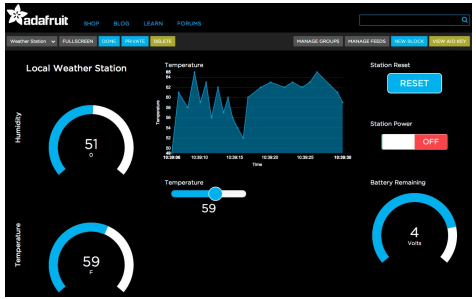

Pub / Sub
Publisher
Publisher
Publisher
Publisher
Publisher
Publishers
MQTT Broker
Subscribers
Subscriber
Subscriber



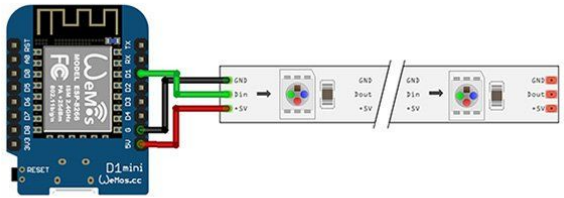

Publish: /feeds/temperature

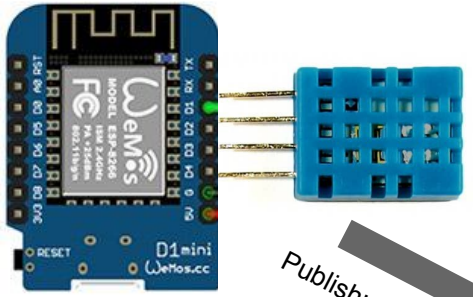
MQTT (BROKER)

Subscribe:
/feeds/temperature




Subscribe: /feeds/temperature

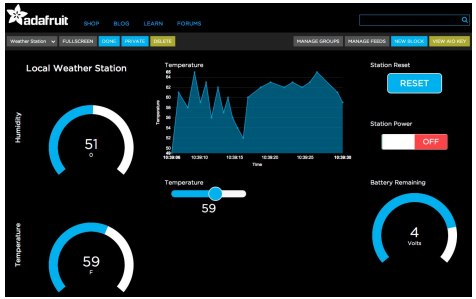




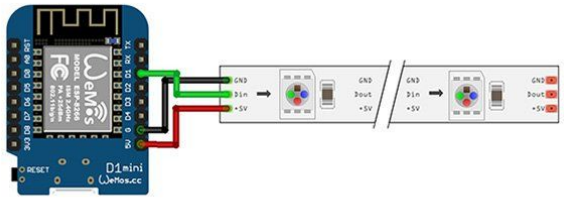

Publish: /feeds/temperature

MQTT (BROKER)

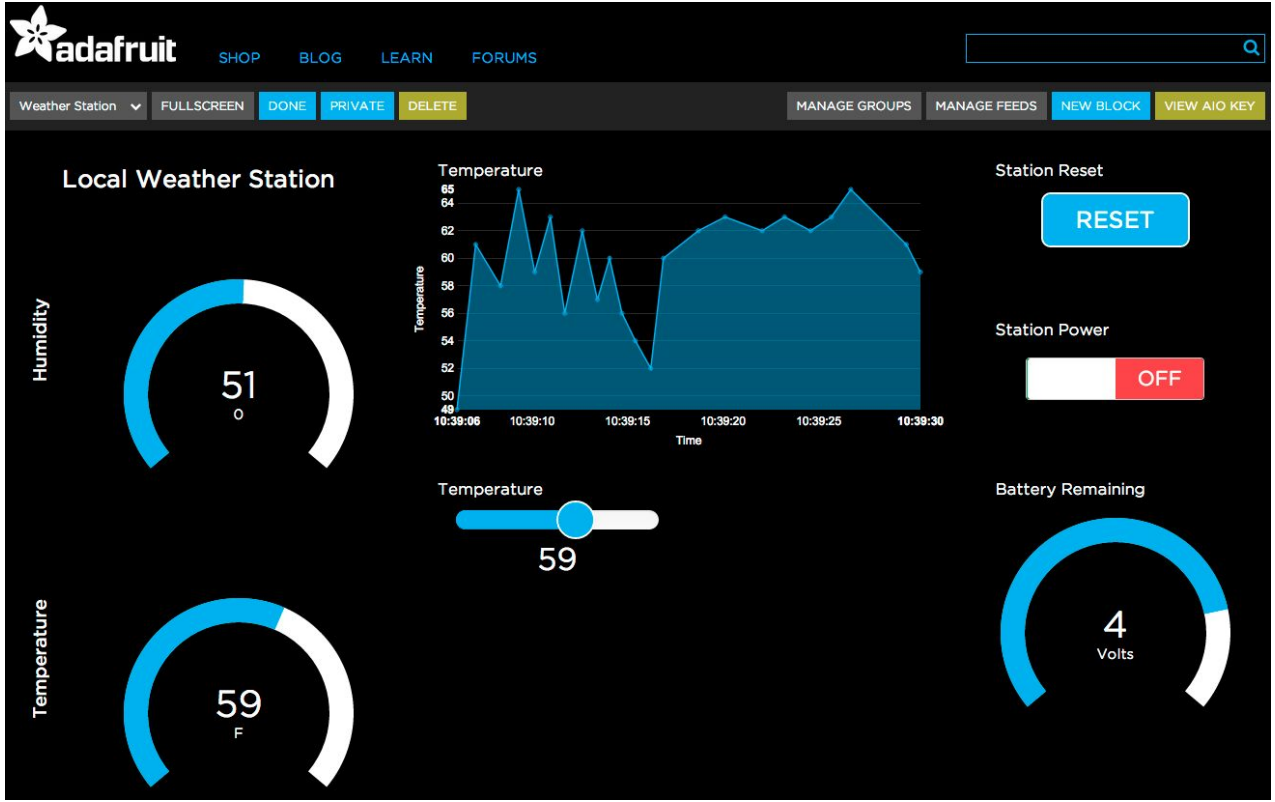
Subscribe: /feeds/temperature




Subscribe: /feeds/temperature




Usually only one publisher pr. feed.




https://learn.adafruit.com/home-automation-in-the-cloud-with-the-esp8266-and-adafruit-io/introduction